User Manual

RapidSSH™

For RapidSSH Version: 1.83

Date: 20 August 2012

1.1 Table of Contents

1.1 Table of Contents 2

2 Introduction to RapidSSH 4

2.1 What is the RapidSSH™ App? 4
2.2 RapidSSH App Requirement 4
2.3 Terminal Overview 5

3 RapidSSH Main Settings 5

3.1 Terminal Settings 6
3.1.1 Columns (Default: 80) 6
3.1.2 Rows (Default: 24) 6
3.1.3 Scrollback (Default: 100) 6
3.1.4 Clear On Connect 7
3.1.5 LineWrap 7
3.1.6 Text Size 7
3.1.7 Console Logging (Default: Off) 7
3.1.8 Auto Upload Logs (Default: Off) 7
3.1.9 Advanced Terminal Settings 8
3.2 Script Manager 10
3.2.1 Create New Script 11
3.2.2 Wait for String Action Item 12
3.2.3 Send String Action Item 13
3.2.4 Terminate Script Action Item 13
3.2.5 Send Hex Bytes Action Item 14
3.2.6 Pause Script Action Item 14
3.2.7 Display Message Action Item 15
3.2.8 Start and Stop Logging Action Item 15
3.2.9 Upload Log Action Item 15
3.3 Keyboard Settings 16
3.3.1 Backspace Key 16
3.3.2 Enter Key 16
3.3.3 Use Option as Ctrl 17
3.4 Dropbox Settings 17
3.5 SSH Settings 18
3.5.1 Keyboard Auth (SSH) 18
3.5.2 Protocol 18
3.5.3 SSH Compression 19
3.5.4 SSH Agent 19
3.5.5 Private / Public Keys 19
3.6 Sharing Settings 24

3.6.1 Remote Server *25*
3.6.2 Private Server *26*
3.6.3 Secure Connection (Default: Off) *26*
3.6.4 Username *27*
3.6.5 Password *27*
3.7 About RapidSSH 27
3.7.1 Version *27*
3.7.2 Copyright *27*
3.7.3 About RapidSSH *27*
4 Launching Connections (Session Manager) 27
4.1 Quick Connect 29
4.2 Connection Manager 29
4.2.1 Creating New Connections via Connection Manager *30*
4.2.2 Managing Connections in Folders *31*
4.2.3 iExplorer for Connection Manager *32*
4.3 In Session Options 33
4.4 Session Sharing 33
4.4.1 Start Session Sharing *34*
4.4.2 Remote User Access *34*
4.4.3 Stop Sharing Session *36*
5 Terminal Features 36
5.1 Terminal Features 37
5.1.1 Keyboard Control *37*
5.1.2 Keyboard Popup Bar Selector *37*
5.1.3 Command Manager *37*
5.1.4 Password Manager *39*
5.1.5 Clipboard Viewer *39*
6 RapidSSH File System 41
6.1 RapidSSH File Types 41
6.2 User Created Text Files 41
6.2.1 Importing files from RapidSSH.com *42*
6.2.2 Importing files from Dropbox account *43*
6.3 Log Files 44
6.3.1 Log File Naming *44*
6.3.2 Uploading Log Files *44*
6.3.3 Auto-Uploading Log files *45*
6.4 .Script and .Connection RapidSSH Files 45
7 General Troubleshooting 46
7.1 Session Sharing Issues 46

2 Introduction to RapidSSH

2.1 What is the RapidSSH™ App?

RapidSSH is a full featured terminal application for Apple iPad's and iPhone's that allows IT engineers to connect and administer remote servers and other devices using SSH, Telnet or Raw protocols via WIFI and 3G.

- The most complete SSH implementation available for Apple iOS – while others support only SSHv2, RapidSSH supports SSH version 1, version 2, SSH Agent, and complete key management using either OpenSSH or Putty formatted keys.
- Full Telnet, and RAW connectivity over WIFI/3G
- VT100, xterm and many other common Terminal emulations
- Encoding support for ASCII, UTF-8 and 17 other major encoding formats, including multibyte characters for non-latin text terminals
- Multiple sessions run concurrently in tabs, including when the App is in the background for upto 10 minutes.
- Saved session import from SecureCRT® and PuTTY via Windows tool and iTunes
- Full Session Scripting using simple yet powerful "Expect X, then Send Y" type scripting.
- One Tap Secure Screen Sharing allows a remote web user to view and interact with the iPad/iPhones terminal window
- Cloud connectivity to both RapidSSH.com and Dropbox.com for dynamically downloading configuration files and scripts and uploading session log files
- Tight integration with iPhone/iPad clipboard and cut/copy/paste directly from the terminal window
- Comprehensive logging support, Command Shortcuts, Stored Passwords, Bluetooth Keyboard and many other features.

And all for a price less than iSSH or Prompt!

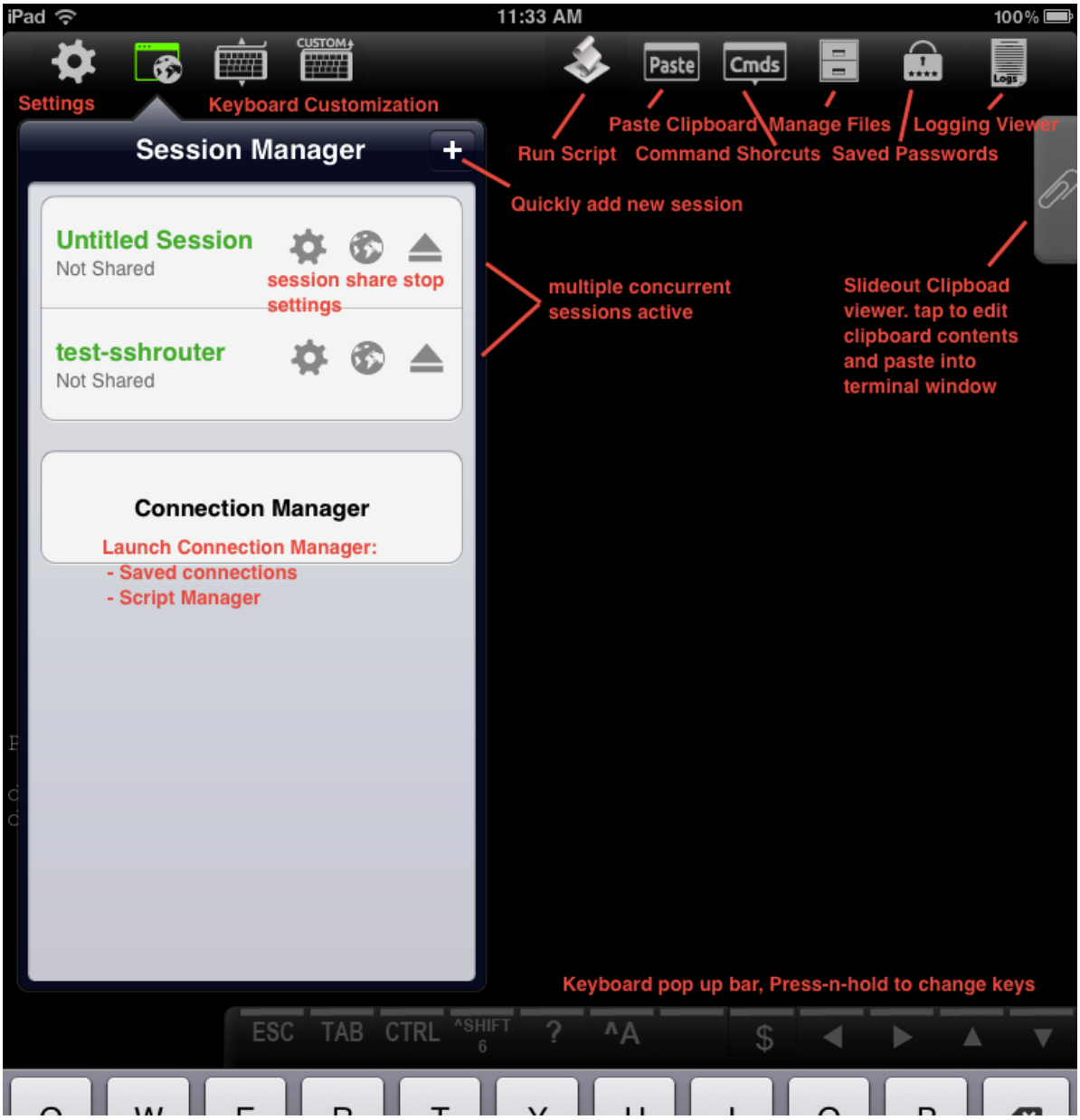
With the RapidSSH app, IT engineers can perform IT equipment maintenance, troubleshooting and disaster recovery operations on remote servers over Wifi/3G using the instant on, portable iPad, iPhone or iPod Touch.

2.2 RapidSSH App Requirement

- iPhone3 + (iOS version 4.2 or later is required)
- iPad1 onwards
- Wifi/3G connection for terminal connection and for sharing sessions

2.3 Terminal Overview

The below picture shows the main RapidSSH window and identifies the major UI features.



Clicking on the Main Settings button will where most of the apps features are configured.

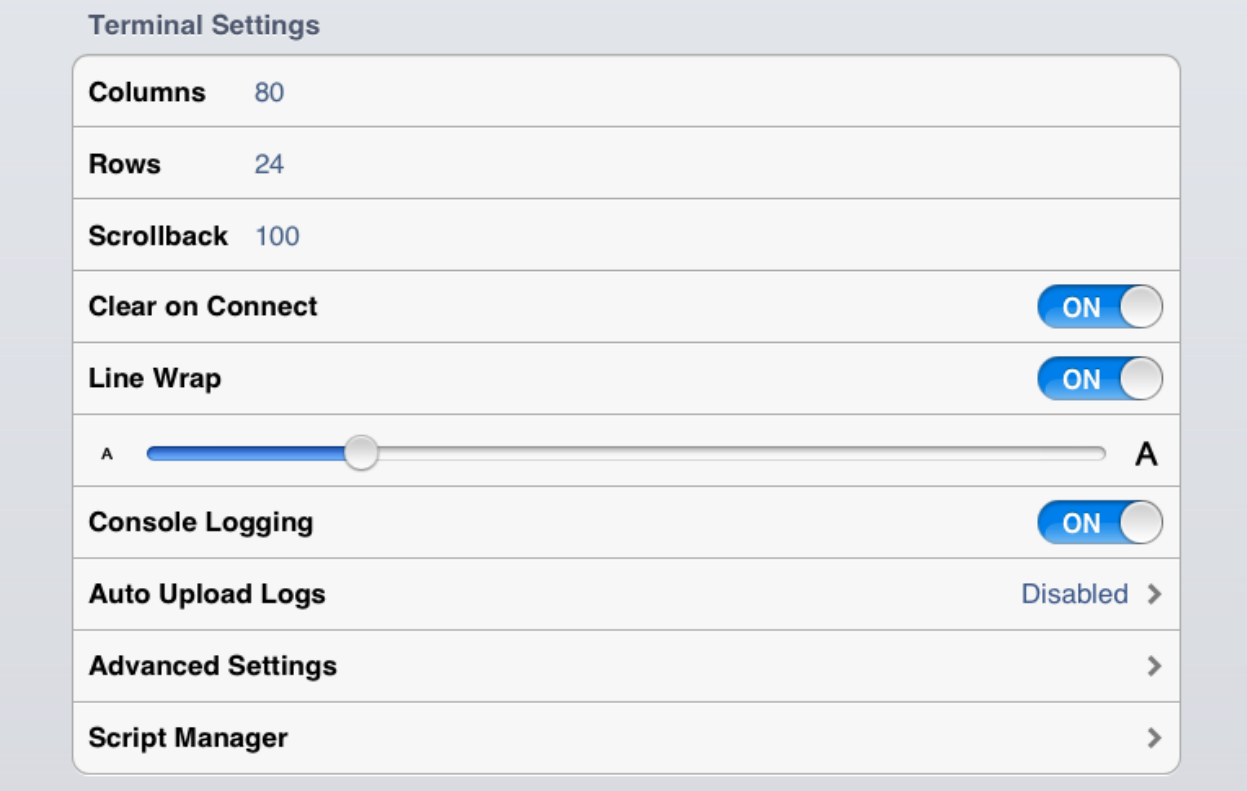
3 RapidSSH Main Settings

The Main Settings page is accessible via the button. This sets the application Default settings for all sessions. Some of these defaults can be varied on individual saved connections (See Section 4 Connection Manager). The App settings cover 5 main areas:

- **Terminal Settings** – features specific to the terminal window – things like linewrap, local echo, scrollback size, but also access to the terminal scripting features.
- **Keyboard Settings** – features for how the keyboard behaves (with or without external Bluetooth keyboard running)
- **SSH Settings** – features specific to SSH Connections (ie Certificates/Keys, Authentication methods)
  Sharing Settings – settings requiring configuration in order to share the terminal window with a remote user via the RapidSSH.com website.

The following section details each of these setting areas.

3.1 Terminal Settings



3.1.1 Columns (Default: 80)

Supports any column width up to 132 characters wide

3.1.2 Rows (Default: 24)

Supports any number of rows on the screen

3.1.3 Scrollback (Default: 100)

Number of lines retained by the console for review. Setting this above 250 will impact performance on iPhone4 and below. iPad2, iPad3 and iPhone 4S supports 500 scrollback without impact due to improved hardware specifications.

3.1.4 Clear On Connect

Defines whether the terminal should clear the screen when starting a new connection

The default is ON

Options:
- **On**
- **Off**

3.1.5 LineWrap

Defines whether the terminal should automatically wrap the line to the next line when the output received from the remote device exceeds the column width and no carriage return has been received.

The default is ON

Options:
- **On**
- **Off**

3.1.6 Text Size

Sets font size in the terminal window

Smaller ---------- Larger

3.1.7 Console Logging (Default: Off)

Logs all printable output to file stored on iOS device. The log files can be uploaded to www.RapidSSH.com website, Dropbox.com folder or extracted via iTunes. Log upload is performed via the Files section described below.

Options:
- **On**
- **Off**

3.1.8 Auto Upload Logs (Default: Off)

Automatically upload log files to remote storage repository when the session finishes. The default is off. Note that this feature will fail if Wifi/3G connectivity is not available at the time the session

Options:
- **Disabled**
- **RapidSSH Remote Server**
- **Dropbox**

Note that to upload logs automatically to a users RapidSSH Public Server, Private Server or Dropbox.com folder requires that the Remote Server username/password be configured in main settings (for RapidSSH Public or Private Servers), or Dropbox account be linked to RapidSSH Application (also in Main Settings).

3.1.9 Advanced Terminal Settings

3.1.9.1 Bell Behavior

Allows the iOS device to create an alert when there is an Terminal event or stay silent.

Options:
- **Play Alert (bell/vibrate phone)**
- **Do Nothing (silent)**

3.1.9.2 Cursor

Allow the choice of cursor style – the default is None

Options:
- **None**
- **Outline**
- **Block**
- **Vertical Line**
- **Underline**

3.1.9.3 Blink Cursor

Cursor blinks – the default is Off

3.1.9.4 Colour Scheme

Changes the colour scheme of the text and background of the console

Options:	
- **White on Black**	White text on Black background
- **Black on White**	Black text on White background
- **White on Black/Grey**	White text on Black/Grey background
- **Green on Black**	Green text on Black background
- **Rainbow**	Helps debugging by colouring different parts of the screen o Blue: Left and right border o Red: Upper border of the scrollback limit o Black: Active area of the console within the scrollback limit

3.1.9.5 ANSI Colour

Allows up to 256 colours on terminal window following the ANSI spec for signaling colour. The default is ON.

Options:
- **On**
- **Off**

3.1.9.6 Local Echo

Controls whether the terminal will echo characters entered to the screen locally or rely on remote serial device to echo them. The default is Auto – where the terminal attempts to recognize if remote serial device is echoing characters, and if so disables local echo. If both local and remote echo are enabled then double characters for every single character typed will appear on terminal window (in which case use “Force Off” setting.

Options:
- **Auto**
- **Force On**
- **Force Off**

3.1.9.7 TERM ENVIRONMENT

Free Text field where Telnet/SSH Sessions can send the TERM=[xxx] to the remote server to indicate capabilities. RapidSSH is a VT100 terminal, however you can configure xterm or other terminal types to improve compatibility with other emulation methods that reject “vt100” terminal settings.

3.1.9.8 Scroll Mode

Defines where the terminal view is positioned. In versions prior to v1.82 there was no change to the terminal view position on input from keyboard or output from remote device. This has changed so the default setting is now “Keyboard Input”.

Options:
- **None – use touch screen to move around the terminal and scrollback window**
- **Keyboard Input (Default) – pressing any key on keyboard will return the window view to the active cursor position within the terminal window**
- **Terminal Activity – any terminal generated activity will center the terminal view to where the terminal activity is being generated**

3.1.9.9 Character Encoding

RapidSSH supports 17 different encoding formats. The default is UTF-8 which supports single and double byte character sets and is the most common character encoding set in modern use. Prior to RapidSSH v1.7 the default encoding set was ASCII which is still available under this setting.

3.1.9.10 Paste Rate Limit

When pasting large text files into the terminal window, the remote device terminal buffer can overflow, producing garbled text.

RapidSSH has a speed control to allow for text from clipboard or scratchpads to be “fed” into the terminal at a slower speed to ensure that the receiving device do not overflow their receive buffers. Internal testing has found that for Cisco telnet sessions, the best setting is “Fast” and hence this is the default. The “Fast” Setting allows for a 800 line configuration file to be pasted into a Cisco device without error in around 10 seconds.

Options:
- **Disabled (Text is sent to active terminal at maximum speed)**
- **Fast (Text is sent in 512byte blocks with 1 second pause between blocks)**
- **Medium (Text is sent in 256 byte blocks with 1 second pause between blocks)**
- **Slow (Text is sent in 128 byte blocks with 1 second pause between blocks)**

3.2 Script Manager

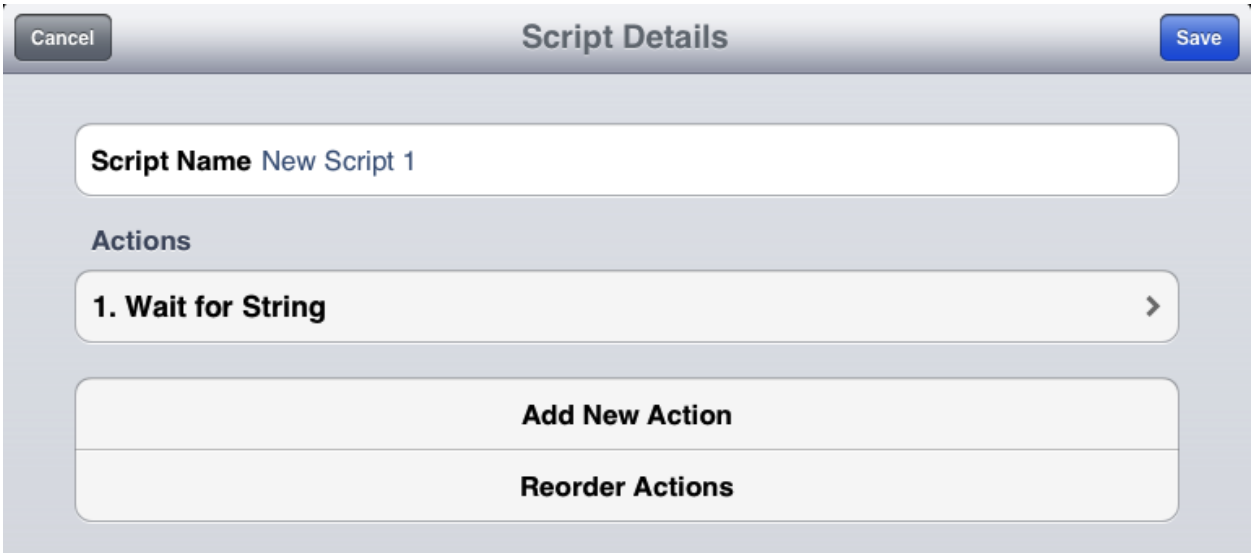
The script manager allows for the creation of terminal scripts. Once created, these scripts can be run immediately on connection by assigning them to saved connection in the Connection Manager, or alternatively by running once a terminal session is started via the Run-Script button on the top bar (iPad) or Tools menu (iPhone).

Scripts are built by adding various action items to the script in order that the terminal should execute them. Usually the script will begin by telling the terminal to wait to see something on the screen (ie the "Wait-for-String" action item) and then the next action item will be to do something (send command, start logging etc). These type of scripts are known as "Expect Scripts" in that while the script is running it will "Expect" to see "X" on the screen, and when it does it will respond with "Y". After creating the script and running it the items are completed sequentially from first item to last.

The below section describes the various action item options are and their configurable parameters. Note that poorly constructed scripts will possibly cause the terminal window to lock or perform unpredictably. A common cause for script problems not configuring a timeout for a Wait-for-String action item – if this is not configured, the script will never progress to the next action item as it has not seen the string it is looking for.

3.2.1 Create New Script

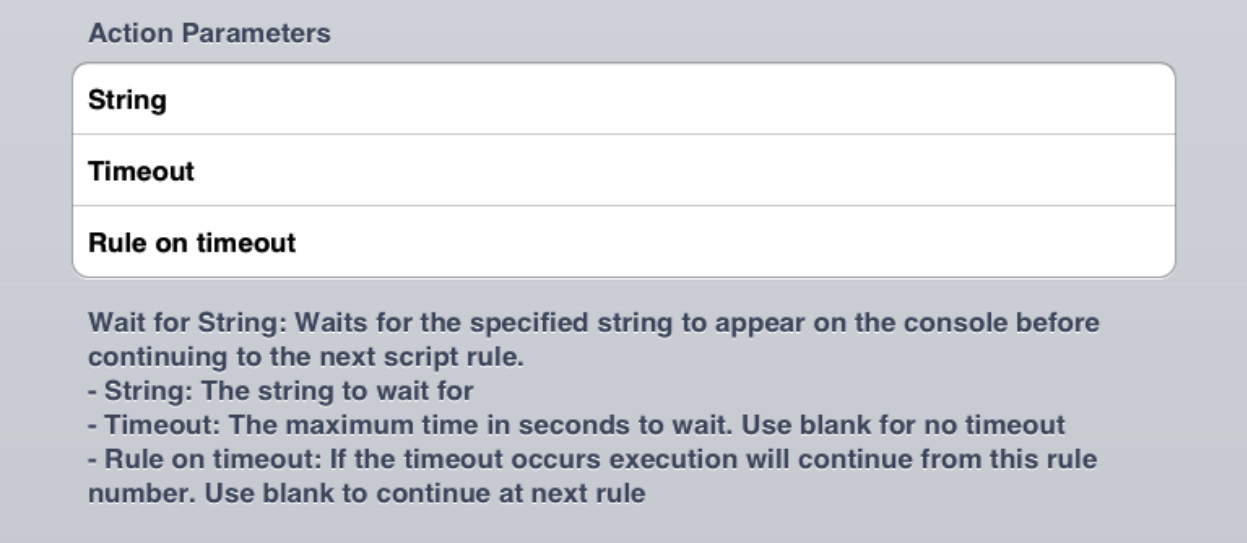
To create a new script, navigate to the Script Manager and tap the Create New Script button. Give the script a name, and then tap on Action 1.



By default the first action is Wait for String, however it can be any of the following action items. Change by tapping item 1 which then allows one of all of the possible action types to be selected.

At the bottom of the screen is where the parameters for the action item is configured, along with some guidance text on which and the type values that can be entered.

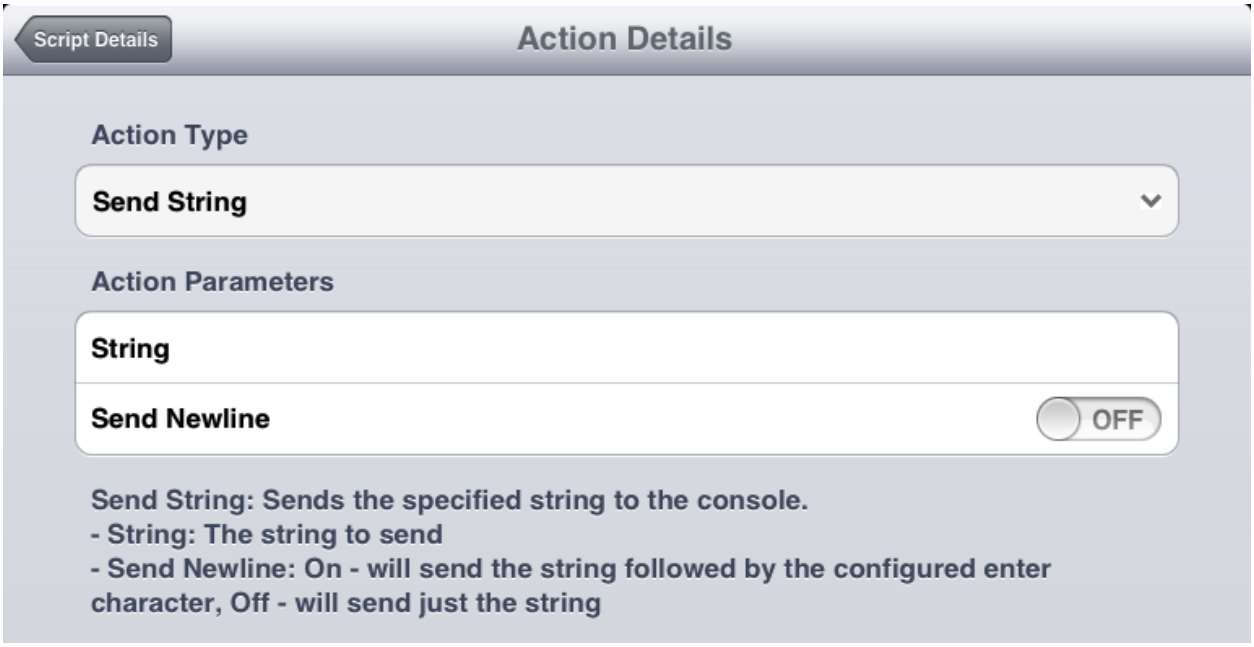
3.2.2 Wait for String Action Item


Action Parameters
String
Timeout
Rule on timeout
Wait for String: Waits for the specified string to appear on the console before continuing to the next script rule.
- String: The string to wait for
- Timeout: The maximum time in seconds to wait. Use blank for no timeout
- Rule on timeout: If the timeout occurs execution will continue from this rule number. Use blank to continue at next rule

Note that the Wait for String action is quite powerful in that it can be used to create loops via the “Rule on timeout” parameter. By default the script executes each item in the script sequentially, however

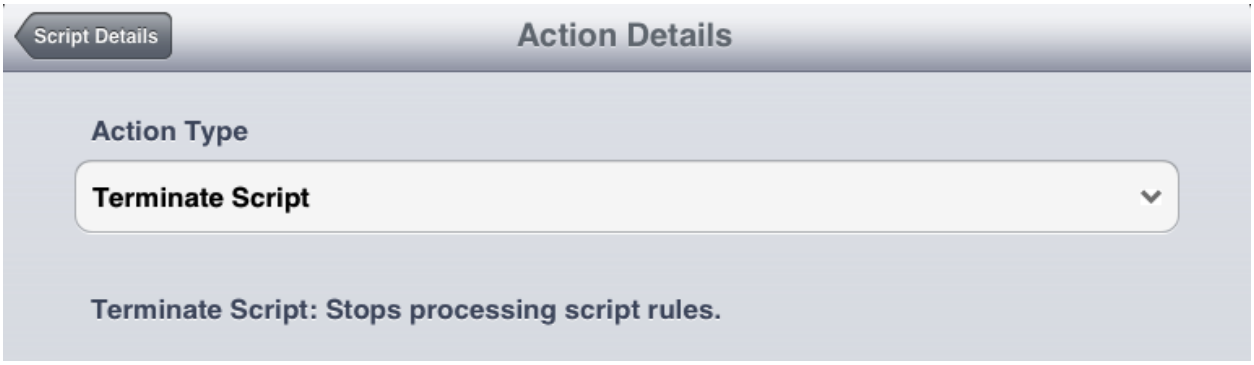
using the Rule on timeout, the script could alternatively jump to earlier or later action items. For example if the script must keep pressing enter until it sees “End of File” then the Wait For String rule is configured to wait to see “End of File”, and if it does not see it after 2 seconds then it jumps back to an earlier “Send String” action item that sends the enter key.

3.2.3 Send String Action Item


Script Details
Action Details
Action Type
Send String
Action Parameters
String
Send Newline
OFF
Send String: Sends the specified string to the console.
- String: The string to send
- Send Newline: On - will send the string followed by the configured enter character, Off - will send just the string

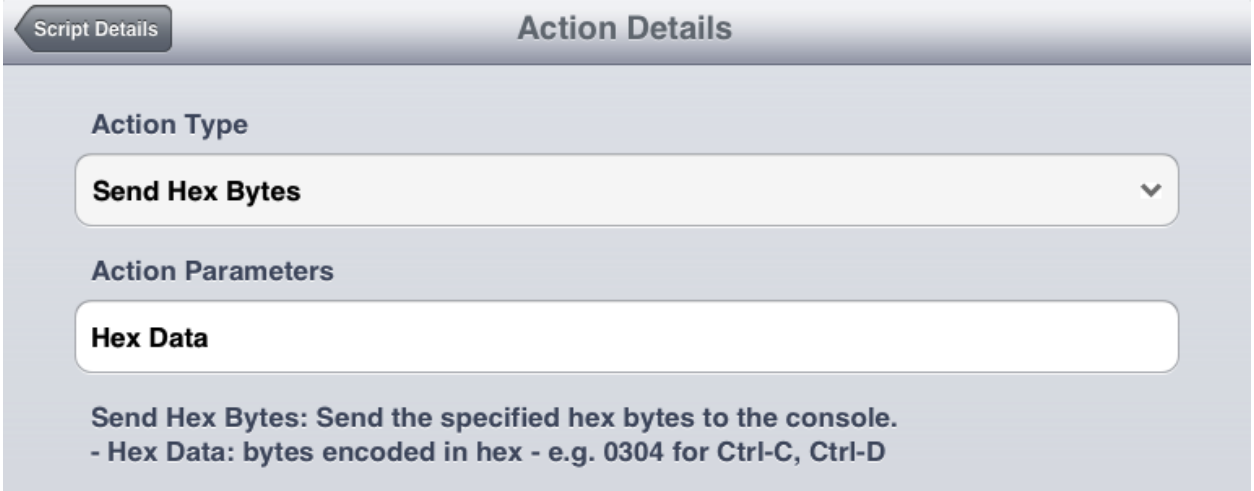
Use the send string to send normal ASCII text to the terminal window, with or without the enter key after it. If escape codes (ie CTRL-X) are needed to be sent, instead use the alternative “Send Hex Bytes” Action item.

3.2.4 Terminate Script Action Item



By default when the script completes its last action item it will terminate by itself, therefore this action is only used where jumps between rules are configured (see Wait-for-String above) and one leg of a script branch requires the script itself to terminate.

3.2.5 Send Hex Bytes Action Item

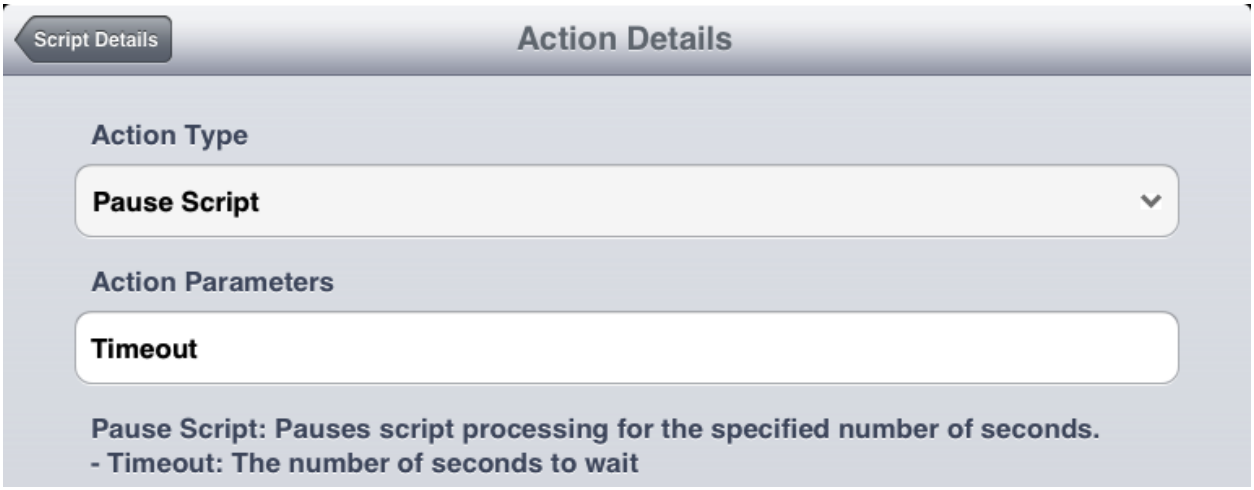

Script Details
Action Details
Action Type
Send Hex Bytes
Action Parameters
Hex Data
Send Hex Bytes: Send the specified hex bytes to the console.
- Hex Data: bytes encoded in hex - e.g. 0304 for Ctrl-C, Ctrl-D

Other common hex byte codes are:

Hex Code	Sends
0D	Carriage Return
09	Tab
08	Backspace

Find more hex byte codes at www.asciitable.com

3.2.6 Pause Script Action Item



Use the Pause action to halt the script for a number of seconds – this is useful when the device connected to will not be ready to accept a “Send String” or “Send Hex Bytes” action immediately after a “Wait for String” has been matched. The pause action is also useful to give the remote device time to process the previous “Send String” before sending the next one.

3.2.7 Display Message Action Item

Display Message is useful to provide feedback to the iPad/iPhone user that the script has completed or is at a certain point. The pop-up message does not have to be acknowledged for the script to continue or complete.

3.2.8 Start and Stop Logging Action Item

Note that these actions override the default Terminal settings for Logging. If Logging is already enabled in the main App settings, there is no need to “Start Logging”, unless it had been previously stopped with the “Stop Logging” Action. These actions are useful if just a subset of terminal data is required to be captured. The parameters allow the log file to be appended to the existing session log file rather than a new one be created.

3.2.9 Upload Log Action Item

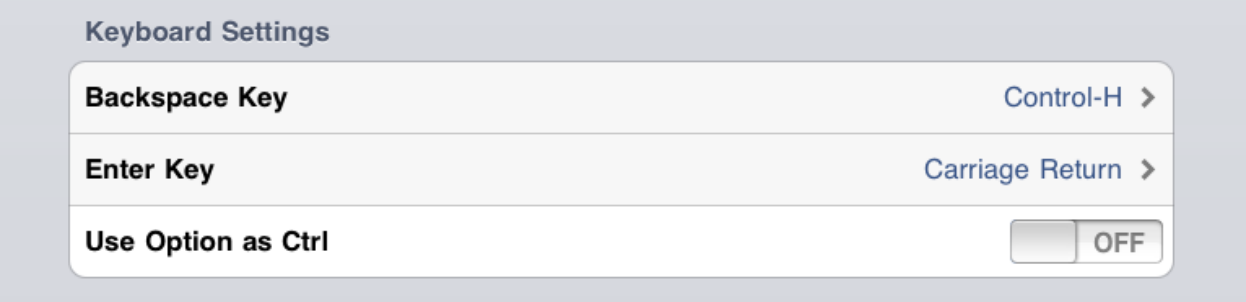
Script Details
Action Details
Action Type
Upload Log
Action Parameters
Upload to
Upload Log: Uploads the console log to the configured remote server. Displays an error if no remote server is configured.

Upload log file created during the session (either because Logging is enabled by default, or due to Start Logging earlier action item). The log file can be either uploaded to the users portal space on the

www.RapidSSH.com website (RapidSSH option), or if the user has linked RapidSSH to their Dropbox.com account then uploaded to the My Apps/RapidSSH/ folder within their Dropbox.

3.3 Keyboard Settings

There are 3 configuration items under the Keyboard Settings section of the main App Settings.



Note that it is possible to use the RapidSSH app with a Bluetooth keyboard and the settings configured here will affect the Bluetooth keyboard. Note also that full Bluetooth Keyboard support is not yet implemented as at version 1.83 – ESC and Arrow keys will not work on Bluetooth keyboards.

3.3.1 Backspace Key

Certain terminals expect Backspace to be sent as CTRL-? (Delete – ASCII 0x7F) rather than the Backspace (CTRL-H – ASCII 0x08). The default setting is CTRL-H.

Options:
- **Control-H**
- **Control-? (127)**

3.3.2 Enter Key

RapidSSH has 3 options for what is sent when the enter key is pressed. By default it sends just Carriage Return (0x0D) which is the OSX standard. This can be changed to Line Feed (0x0A) or both Carriage Return followed by Line Feed (0x0D0A) each time the enter key is pressed.

Options:
- **Carriage Return**
- **Line Feed**
- **CR + LF**

3.3.3 Use Option as Ctrl

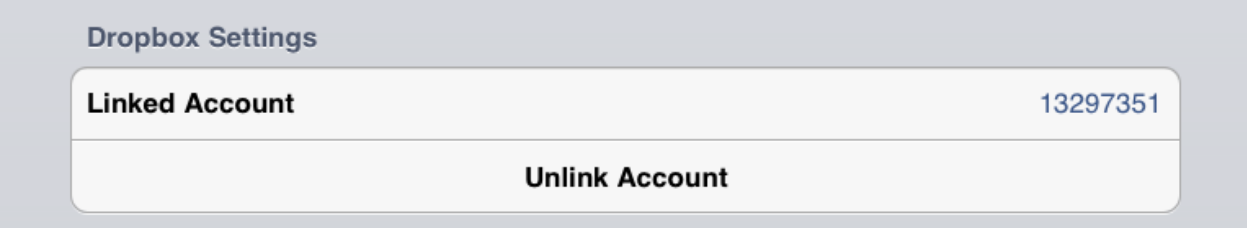
RapidSSH implements a workaround for the control key not working on Bluetooth keyboards. If using a Bluetooth keyboard with RapidSSH, you can select the “Use Option and Ctrl” to use the Alt/Option key on the keyboard to send CTRL-[key] sequences. The default setting is OFF.

Options:
- **On**
- **Off**

3.4 Dropbox Settings

In version 1.82 and later of RapidSSH, the Dropbox.com linking is performed in the Main application settings rather than in the Logs, Files and script manager individual settings pages. To link Dropbox account to RapidSSH, click the “Link Account” button and enter Dropbox.com email and password

To unlink a linked account use the Unlink button. No files are deleted from Dropbox or RapidSSH’s local file system when accounts are linked or unlinked.



3.5 SSH Settings

Unlinke most other iOS terminal applications, RapidSSH has full support for SSH including both SSH version 1 and 2, as well as SSH Agent, Agent Forwarding, Key Generation, Key import and export as well as conversion of key formats between the common PuTTY and OpenSSH key formats. This section describes the main global configuration options available for SSH connections.

3.5.1 Keyboard Auth (SSH)

Allows for the “Keyboard Interactive” password authentication method in addition to the usual password security. Use Keyboard Interactive method to SSH into servers that require this method – for example to SSH to Mac OSX Lion Server requires “Keyboard Interactive” method. Try alternating the method if you can launch and to connect to an SSH server but no login or password: prompt appears. The default method is OFF.

Options:
- **On**
- **Off**

3.5.2 Protocol

RapidSSH supports negotiation of either the SSHv1 or v2 protocol. By default the app will attempt to negotiate SSHv2, but if not available from the server will fall back to SSHv1. These settings can be changed via the following options. Note as at version 1.84 that this is a global configuration change affecting all sessions.

Options:

- **SSH1 Only**
- **SSH1 Preferred**
- **SSH2 Preferred**
- **SSH2 Only**

3.5.3 SSH Compression

RapidSSH supports the negotiation of "zlib" based compression for the SSH connection. This increases performance where a large amount of terminal data is being sent / received especially over 3G connections. Note that the SSH server must support the Zlib compression option for this to be negotiated. The use of compression has a minor impact to performance on iPhone 3GS.

3.5.4 SSH Agent

RapidSSH also supports SSH Agent and Agent Forwarding.

- SSH Agent is a separate part of the RapidSSH app that (when enabled) stores and securely grants access to imported or generated Private Keys, for responding to public-key based authentication challenges issued by remote SSH servers. On first use of a Private Key for an authentication request the user will be prompted for the decryption key (if it is encrypted). All subsequent direct connections (one hop) to servers requiring the same Private Key will be handled automatically without the user being prompted. The SSH agent supports Private Keys in the Putty format, RSA-1 format, and OpenSSH format (unencrypted only)

- SSH Agent Forwarding extends the use of the RapidSSH Agent beyond the first hop SSH server to subsequent servers so that when (for example) a SSH session to Server 1 is then used to create another SSH session from Server 1 to Server 2, the authentication for the Server 2 connection from Server 1 is forwarded all the way back to the RapidSSH Agent running on the iPhone/iPad. Note that Agent Forwarding requires the cooperation of the SSH server and client processes at intermediate hops along the way and may need to be configured (for OpenSSH/sshd based systems, check the AllowAgentForwarding setting in sshd_config, and the ForwardAgent setting in ssh_config on Server 1) The options available in this setting menu are:

Options:
- **Enable SSH Agent**
- **Enable SSH Agent Forwarding**

Note that SSH Agent must be enabled in order to also enable SSH Agent Forwarding.

Once SSH Agent is enabled, individual keys from the SSH Private Keys section must be enabled for use by the Agent (see "Existing Keys" section below).

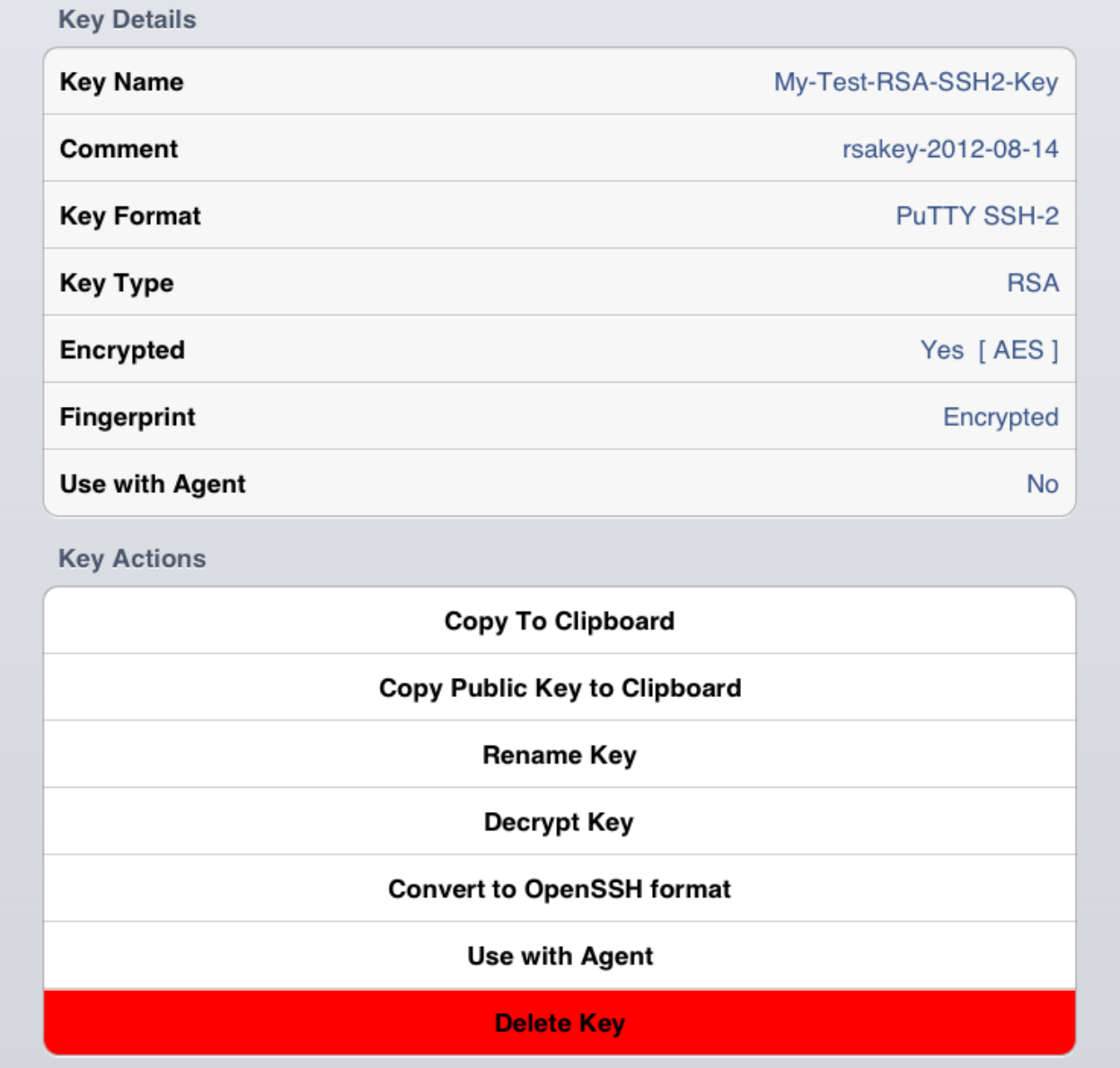
3.5.5 Private / Public Keys

This section allows for the import of Private and the generation of Private and Public key pairs. Once imported and / or generated, the keys can be assigned to SSH connections or permitted for use by the

SSH Agent. In addition, keys imported in one format (OpenSSH or PuTTY) can be converted to the alternative format.

3.5.5.1 Existing Keys:

Displays a list of the available imported or generated keys already in RapidSSH App. By tapping an existing key the Key Detail page for that key will appear with addition options as per below:



The Key Detail screen shows the name and comment associated with the key – often the comment is used to remind user which remote servers the public part of the key pair has been installed on.

Other information described:

Options:	
Copy to Clipboard	Copy to Clipboard – Copies the private keyfile to the clipboard for use in Get Console or other applications (e.g. an email application for sending to a third party).
Copy Public Key to Clipboard	Copies the Public Key portion to the clipboard in a format for installing into remote servers (e.g. ~/.ssh/authorized_keys file). Supports Putty format, RSA-1 format, and OpenSSH format (unencrypted only)

Rename Key	Change the displayed name for the key without changing the key itself
Decrypt / Encrypt Key	Removes / Adds encryption (and passphrase protection) to key
Convert to OpenSSH / PuTTY format	Shift the key format between Putty and OpenSSH versions. In general RapidSSH prefers the PuTTY format, but if imported in OpenSSH format, the key can be used un-converted in SSH Agent so long as it is not encrypted
Use with Agent	Allows the key to be used by the RapidSSH SSH Agent software, thus caching the passphrase used to unlock it, and automatically responding to SSH authentication challenges requiring it.

The operations possible with the shown key are described in the below table

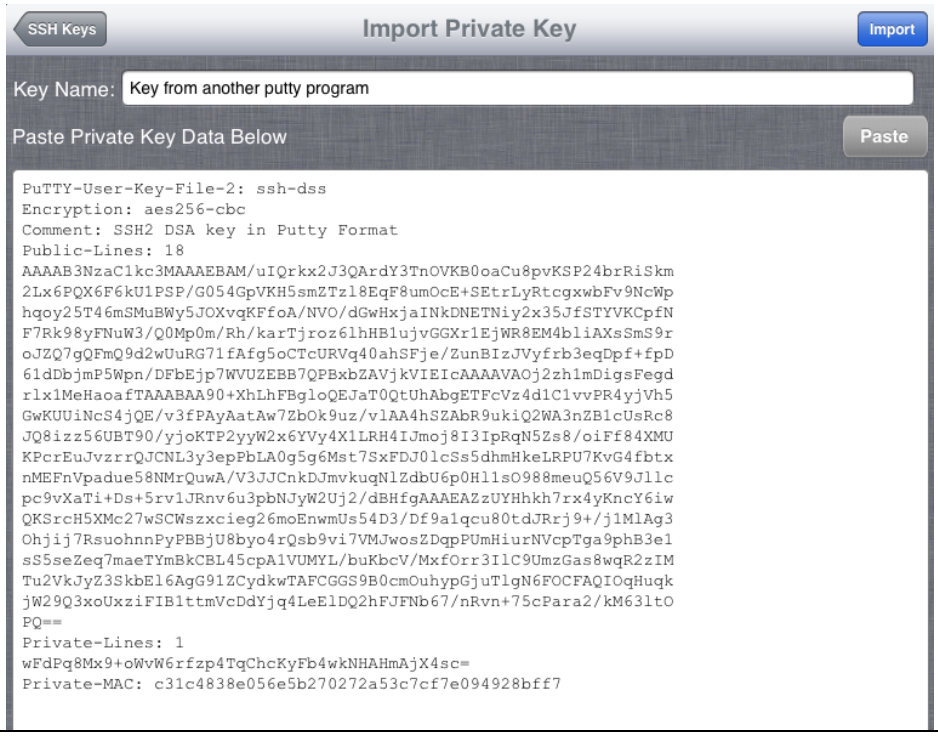
Options:	
Key Format	OpenSSH or Putty Format. Note that while the public key format is standardized, the Private Key can take either PuTTY or OpenSSH format. Rapid SSH generates key pairs in the PuTTY format, and can use keys imported in the OpenSSH Format natively or via conversion
Key Type	RSA (SSH2), RSA (SSH1), or DSA (SSH2)
Encrypted	If the Key has itself been encrypted with a passphrase (either AES or 3DES encryption), else not encrypted
Fingerprint	Displays the fingerprint of the key (if available). The fingerprint is available for Putty format, RSA-1 format, and OpenSSH format (unencrypted only)
Use with Agent	Shows if this Key can be used by RapidSSH's built in SSH Agent

3.5.5.2 Import Key:

Paste the OpenSSH or Putty formatted key data into this window then press "Import" button. If successful a dialog box will appear confirming import and the key will be available to use / assign to saved SSH connections.

Options:

Key Name	Give imported key a name for identification in Key Manager
Paste Private Key Data Below	Press “Paste” button to put contents of clipboard into paste window, then click import. RapidSSH will detect the key format and key type and move to Key Details page for the imported key. The below example shows the key data for a PuTTY formatted DSA Key.

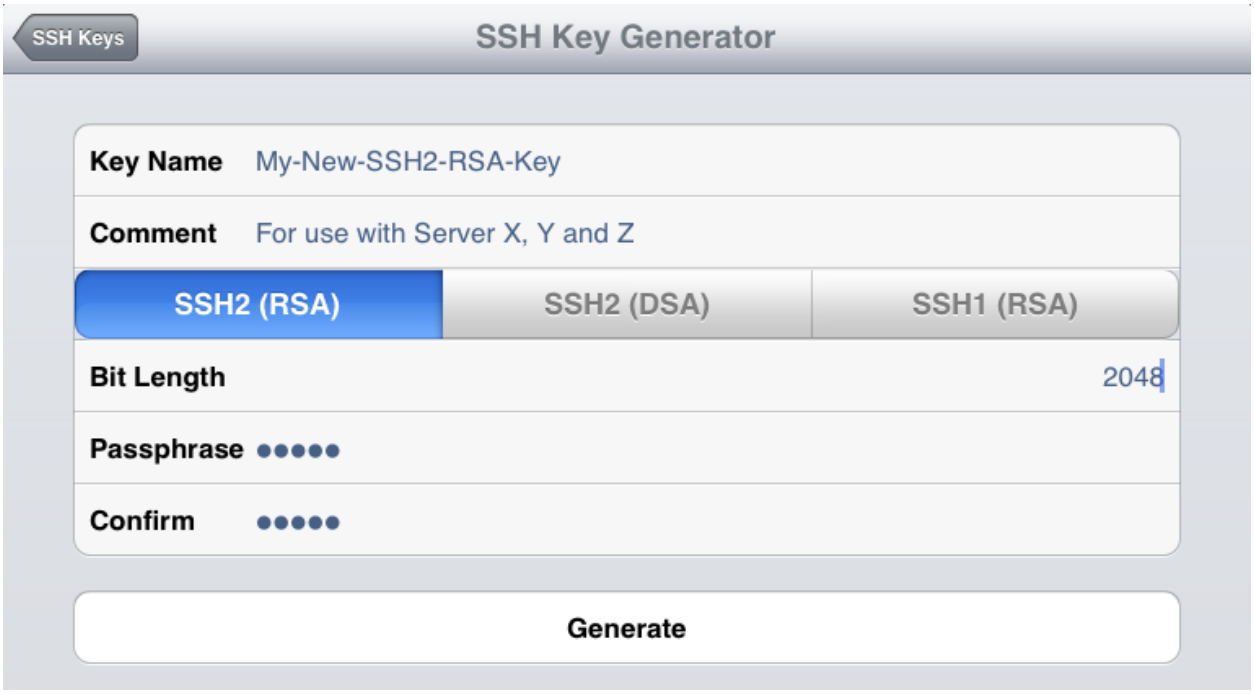


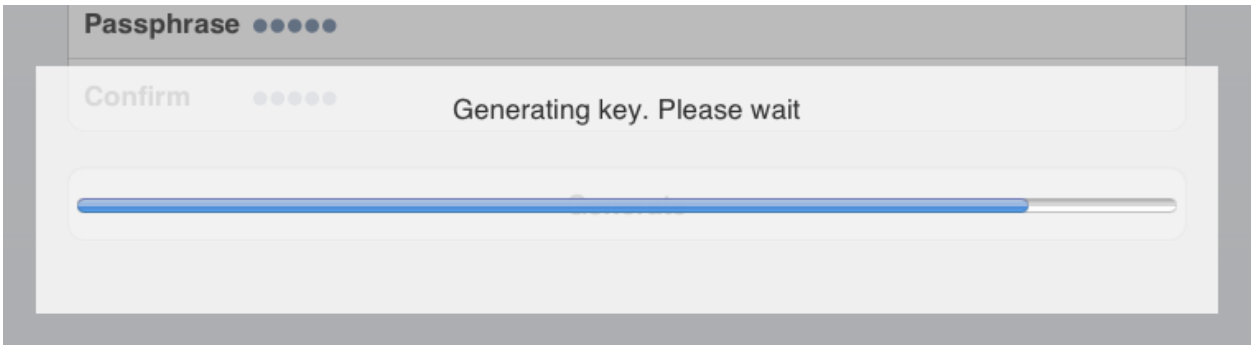
3.5.5.3 Generate Key:

RapidSSH can generate Putty formatted keys upto 2048 bits in length. The following options are available as per description below:

Options:	
Key Name and Comment	Enter a name and comment to identify the generated key
Key Type	RSA (SSH2), RSA (SSH1), or DSA (SSH2)
Bit Length	Upto 2048 bits in length in multiples of 512 bits
Passphrase	Encrypt the generated key with a passphrase. This passphrase will be needed when the key is used to login to a server, or by the SSH agent on its first use of the key. The passphrase is optional but highly recommended. Leave this field blank for no passphrase.
Confirm	Enter the passphrase again

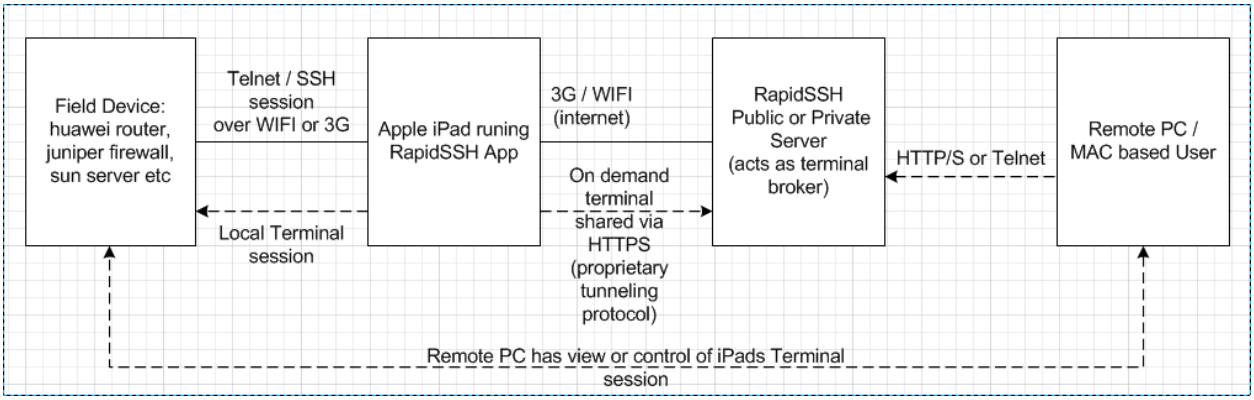
The below screen shots show valid entries for the Key Generator.



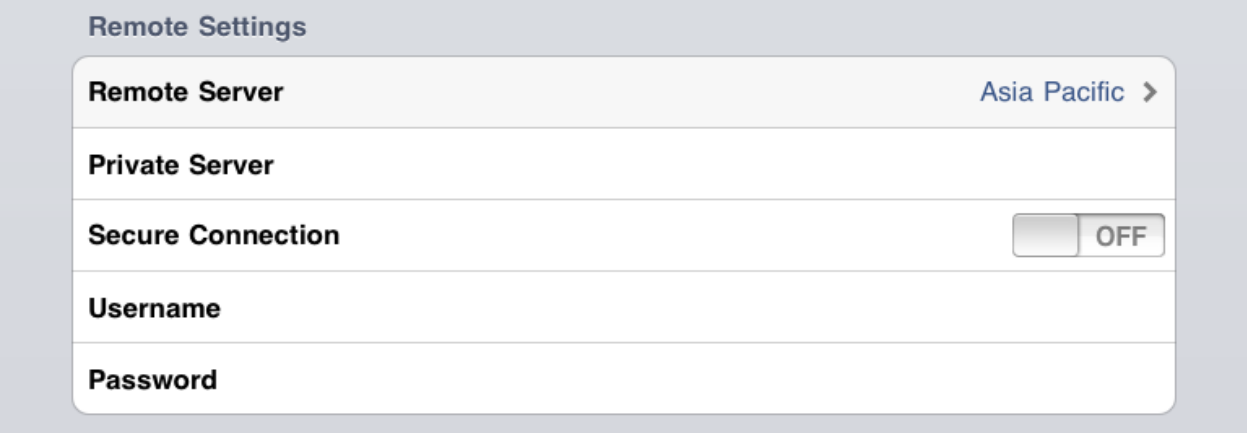


3.6 Sharing Settings

RapidSSH allows you to share your terminal window(s) with remote web users. Each concurrent terminal session on the iOS device can be shared independently with the RapidSSH.com website, and made accessible via individual one-time token codes. The below drawing provides a simple overview of how the session sharing feature works:


Field Device: huawei router, juniper firewall, sun server etc
Telnet / SSH session over WIFI or 3G
Local Terminal session
Apple iPad runing RapidSSH App
3G / WIFI (internet)
On demand terminal shared via HTTPS (proprietary tunneling protocol)
RapidSSH Public or Private Server (acts as terminal broker)
HTTP/S or Telnet
Remote PC / MAC based User
Remote PC has view or control of iPads Terminal session

In order to enable session sharing, the "Sharing Sessions" details must be populated in the App settings.



3.6.1 Remote Server

Select the closest Public server to the iPad/iPhone user for session sharing with a remote user, or if Private Server has been installed then select Private Server.

Public versions of the Server are hosted in US, UK and New Zealand. These can be used at no charge by any registered purchasers of the RapidSSH App to allow remote web access the terminal session via the public website www.RapidSSH.com. The performance of the end to end terminal varies greatly on the current load of the server and the latency of both the Apple device and the Remote web users from the selected public server.

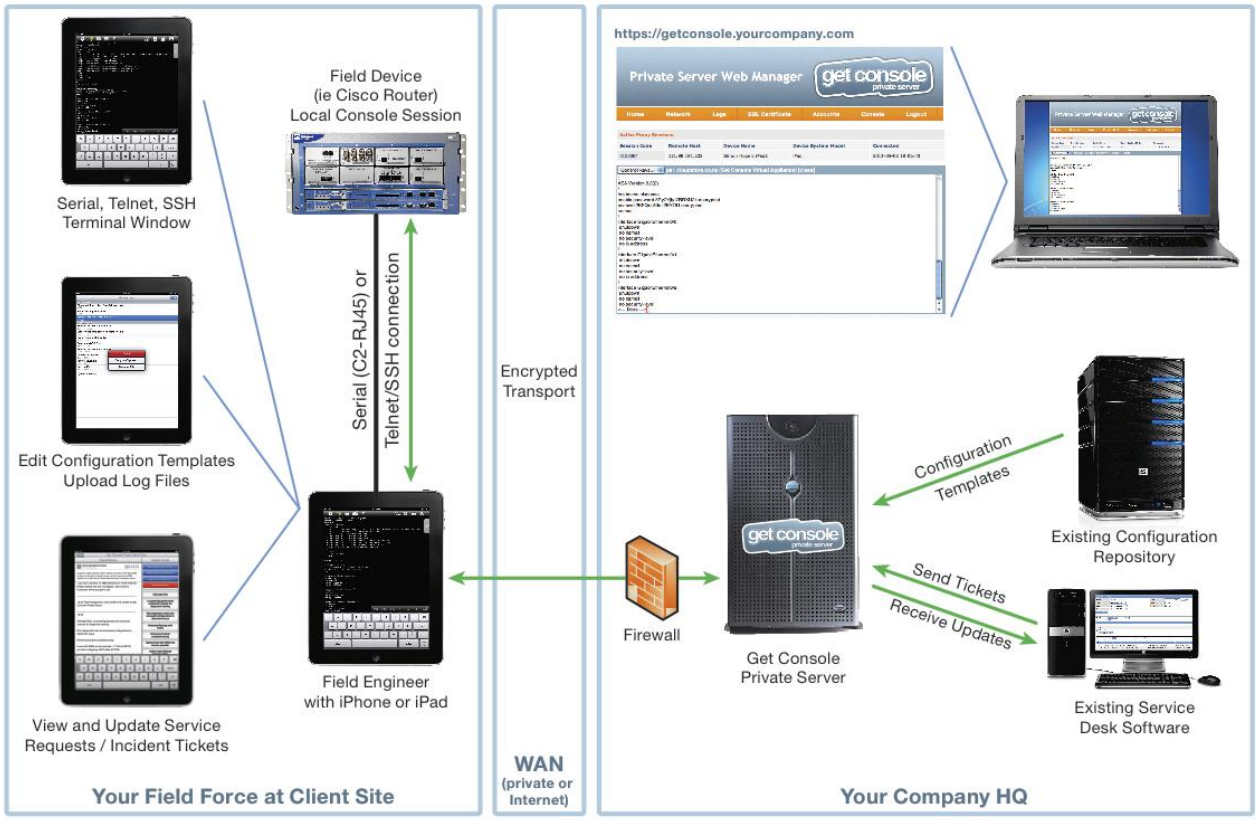
Options:	
- **Asia Pacific**	
- **Europe**	
- **North America**	
- **Private Server**	The Private Server is hosted on a customer's own network, and secured by their own network security policy
- **Disabled**	The option to share remote session will be disabled

3.6.2 Private Server

Private Server is an optional server side component to RapidSSH that allows for end users to host their own RapidSSH server on their own network and infrastructure.

If Private Server is selected for Remote Server, set its hostname or IP address in this field.

Below is diagram representation of the working of the Private Server



For more information, please view the RapidSSH Private Server manual available from www.RapidSSH.com/private-server

3.6.3 Secure Connection (Default: Off)

Use SSL for session sharing for more secure connection. SSL option should only be selected with Private Server if the Private Server has a valid and publicly trusted (ie trusted natively by iPhone / iPad) SSL certificate for the entered hostname.

Enabling Secure Connection has a small impact on the remote performance. For maximum responsiveness when encryption is not required leave Secure Connection OFF.

Options:
- On
- Off

3.6.4 Username

Registered email address on www.RapidSSH.com – it's free to register. Or if Private Server has been chosen, the user name for use with the Private Server login

3.6.5 Password

Password for your registered account on www.RapidSSH.com website. Or if Private Server has been chosen, the password for use with the Private Server login.

3.7 About RapidSSH

3.7.1 Version

Version number of the RapidSSH App. Certain features are only available in later versions. Updates to RapidSSH occur automatically via the App Update process in the iTunes App Store.

3.7.2 Copyright

Year and holder of the copyright. RapidSSH is a registered trademark owned by Cloudstore Limited.

3.7.3 About RapidSSH

About RapidSSH app – Displays version, copyright information, developers, LibSSH2 license and any other licenses for third party software that is incorporated into RapidSSH.

4 Launching Connections (Session Manager)

Pressing the Connection button will launch the new connection Session Manager popup (iPad) or full screen Session Manager (iPhone). RapidSSH provides 2 ways to launch connections – first via the plus button for a Quick connection, or via the Connection Manager to launch a saved or recent (unsaved) quick connection. The below drawing highlights the main features of the Session Manager.

As can be seen, the Session Manager popup shows all active and also recently closed sessions that can be restarted. Within each session the options (ie like scrollback length, window size etc) can be varied from the Main Settings defaults via the session options button. Any live session can be shared a remote user via the “world” button and the session can be stopped via the “eject” button.

If a session is stopped or times out, then it still remains visible in the Session Manager until it is deleted via the X button or is restarted via the restart button.

There are hardware limits to the number of concurrent sessions that can be run in RapidSSH, depending on the Apple iOS device in use. The below table provides these limits:

Apple Hardware	Number of Concurrent Live Sessions Allowed
iPhone 3G	3
iPhone 3GS	5
iPhone 4	8
iPhone 4S	15
iPad 1	3
iPad 2	8
iPad 3	20
iPod Touch 3rd Gen	3
iPod Touch 4th Gen	8

4.1 Quick Connect

Pressing the "+" button on the Session Manager launches the Quick Connect popup. This dialog allows for the instant launching of a new Telnet, SSH, or Raw connection in addition to launching an automatically saved recent connection (Either Quick connected or launched from the Connection Manager).

Depending on the type of connection, various session parameters (such as hostname/IP address) will need to be completed and then tap the Connect button.

4.2 Connection Manager

The Connection Manager is used to create, modify and launch saved connections. Tapping Connection Manager in the Session Manager popup will launch Connection Manager where all saved connections can be administered.

Creating new connections can be done one of 3 ways

1) Via the Connection Manager "New Connection" dialog box (discussed in this section)

2) By converting a Recent Quick Connection to a saved connection via the blue arrow (Give the recent connection a Description and then tap Save)
3) By importing ".connection" files into RapidSSH via iTunes. RapidSSH provides a free tool to convert existing PuTTY and SecureCRT .ini files into RapidSSH ".connection" files. This tool can be downloaded from www.RapidSSH.com/tools . The method to import the ".connection" files into RapidSSH via iTunes is documented at www.RapidSSH.com/tools/importutility/

4.2.1 Creating New Connections via Connection Manager

4.2.1.1 SSH

For creating new SSH connections, complete the following items

Description: Define a name for this connection

Hostname: Enter hostname or IP address of the device to connect to via SSH
Port (Default: 22): Enter port number of the device to connect to via SSH
Username: Enter username to login to the device
SSH Key: Choose an SSH key to login to the device if using certificate based authentication. The default is None. If SSH Agent is enabled for this Key, then this will only be required for the first connection.

Terminal Settings: This section allows for the override of the Main Settings default Terminal Settings to apply to this connection. Change any of the allowed settings.

Click Save when complete

4.2.1.2 Telnet

For creating new Telnet connections, complete the following items

Description: Define a name for this connection
Hostname: Enter hostname or IP address of the device to connect to via Telnet
Port (Default: 23): Enter port number of the device to connect to via Telnet. Note that if a non-standard port is chosen, the Telnet preamble will still be sent. If a connection over a non standard port is required without the Telnet preamble (ie for when "Telnetting on port 80 to a webserver" to check that it responds

4.2.1.3 Raw

For creating new Telnet RAW connections, complete the following items

Description: Define a name for this connection
Hostname: Enter hostname or IP address of the device to connect to via Telnet
Port (Default: 23): Enter port number of the device to connect to via Telnet. Note even if port 23 is selected the Telnet preamble will NOT be sent. Common use of Telnet RAW is to test-open connections to web servers and mail servers to see if they are responding on their standard port 80 / port 25

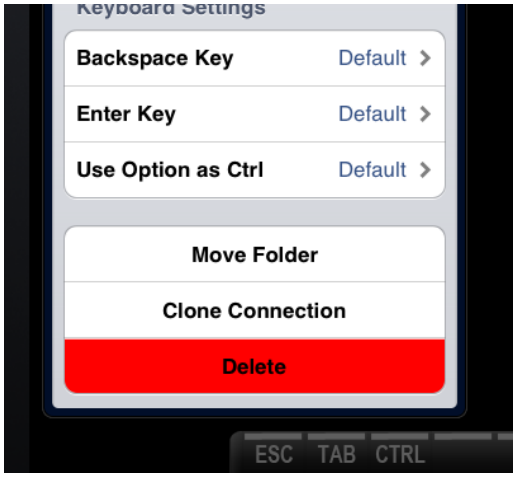
4.2.2 Managing Connections in Folders

Connection Manager allows for the creation of connections, grouping of connections into subfolders and also for cloning connections and moving them between subfolders. These folders are stored in the RapidSSH Application's file system area within the iPhone/iPad. They can be exported via iTunes, however to import connections into a specific subfolder is not possible via iTunes. To perform this manipulation outside of Connection Manager requires a third party application. We have tested one such application called iExplorer which is discussed below.

4.2.2.1 Creating and Managing folders and sub-folders for Connections

Tap "New Folder" to create a new folder and enter name in dialog box. The folder appears at the top of the Saved Connections List. Folders can be nested, so to create a subfolder navigate to the new folder by tapping it then tap the New Folder button again.

Saved connections can be moved from one folder to another or to the root. To move a connection tap the blue arrow next to a saved connection, scroll to the bottom where there are options to Move (to a) Folder, as well as Clone the Connection or Delete.



A sub-folder can also be created under another folder with connections in it (see "Test Connections Folder" in above screenshot). Note that if you delete a folder which contains a connection, the connection will be moved to the root folder instead of being deleted together with the folder.

4.2.2.2 Recent Connections

RapidSSH Connection Manager presents the 10 most recent connections below the Saved Connections. The connections listed in Recent connections include Quick Connections and Saved Connections. This list can be cleared by tapping the "Clear Recent Connections" button at the very bottom of the list. Alternatively, a recent connection that was made via "Quick Connect" method can be saved to the Connection Manager via tapping the blue arrow, providing a name (description) for the connection and then tapping save. The connection will then appear in the root Saved Connections folder of the Connection Manager.

4.2.3 iExplorer for Connection Manager

As mentioned in section 4.2 the Connection Manager content can also be populated or edited via your PC/Mac by a free program called iExplorer. Cloudstore makes no assurances about the stability and offers no support for iExplorer, however in our internal testing we have found the current version to be useful for bulk arranging connection files within folders and subfolders outside of the RapidSSH app.

iExplorer (formerly named "iPhone Explorer) can be downloaded from http://www.macroplant.com/iexplorer/ currently free of charge.

4.2.3.1 iExplorer Use

After the iExplorer is installed in your PC/Mac, follow these simple steps to create/edit/delete your own connections:

1. Ensure you have a suitably late version of iTunes installed
2. Connect your iPhone/iPod/iPad to the PC/Mac using its cable
3. Open iExplorer, navigate to the iDevice -> Apps -> com.amixcapital.rapidssh/Documents/Connections/[Folder]

a. **To edit a connection:** Drag and drop any .connection file into your computer to edit its content using a text editor e.g. Notepad, Notepad++, you can then copy the file back to its original location to overwrite it with the new information
b. **To add new connection(s):** Create a new .connection file in your computer then drag and drop it into iDevice -> Apps -> com.amixcapital.rapidssh/Documents/Connections/[Folder]
c. **To delete connection(s):** Delete a .connection file from the iDevice -> Apps -> com.amixcapital.rapidssh/Documents/Connections/[Folder]

When adding or editing a connection, simply edit the .connection file accordingly using xml protocol.

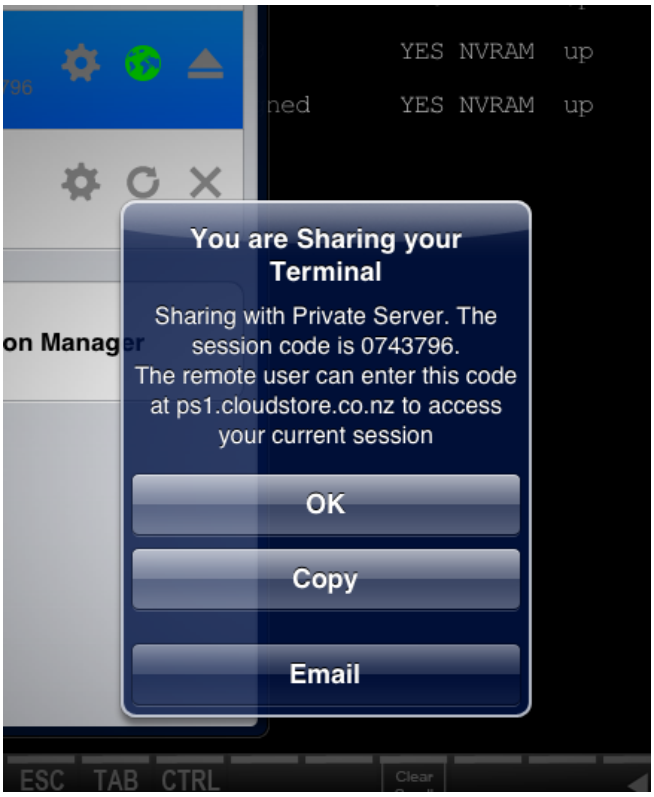
4.3 In Session Options

Once a connection has been established (or attempted to be established) it will appear in the Session Manager. From this popup the actions that can be performed depends on whether the connection is live (connected - green) or closed (disconnected - red)

The following table describes the actions available.

	Session Options: Allows for editing of changeable settings while the session is live, for example changing the terminal width, scrollback, colour scheme etc)
	Share Session: Will share the terminal screen with the configured remote server (Public or Private), and generate a one-time token code. See the session sharing Settings (section 3.6 above) and how to operation below.
	Disconnect Session: Disconnect from the remote server (Telnet/SSH/Raw) The session will stay visible in session manager list of sessions with the 2 below options available.
	Restart Disconnected Session: Attempts to reconnect the session to the remote server (Telnet/SSH/Raw)
	Remove Disconnected Session from Session Manager

4.4 Session Sharing


YES NVRAM up
YES NVRAM up
You are Sharing your Terminal
Sharing with Private Server. The session code is 0743796.
The remote user can enter this code at ps1.cloudstore.co.nz to access your current session
OK
Copy
Email
ESC TAB CTRL

Tapping the button will attempt to share the selected terminal session with the RapidSSH website or configured Private Server so that a remote web user can see and interact with the terminal session at the same time. In order to use this feature the settings for session sharing must be configured in the main settings page (see section 3.6 above).

Each shared terminal session is dynamically given a one-time token code at the time session sharing is initiated. This code is used to secure access to the users iPad/iPhone. The remote web user must know this 7-digit token code in order to see the iPad/iPhones terminal window.

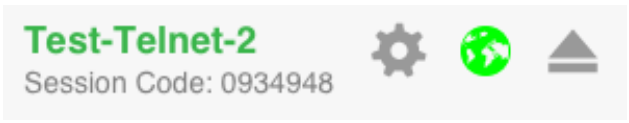
Only sessions that are shared are visible to remote users, however multiple concurrent sessions can be shared – each with their own unique one-time token code.

4.4.1 Start Session Sharing

Once the notification "You are Sharing your Terminal" appears, the remote user can access the iPad via the token code displayed in the notification. To make it easy for the remote user to learn the code it can be emailed or copied to clipboard via this notification window.

Options	Function
- **OK**	Return to the active terminal window
- **Copy**	Copy the remote access token code to the clipboard so that it can be used in txt message or other iOS device message
- **Email**	Email the remote access token code to a recipient with instructions how to connect to the shared session

Once shared, the token code will also appear in the Session Manager, and the globe icon will show green

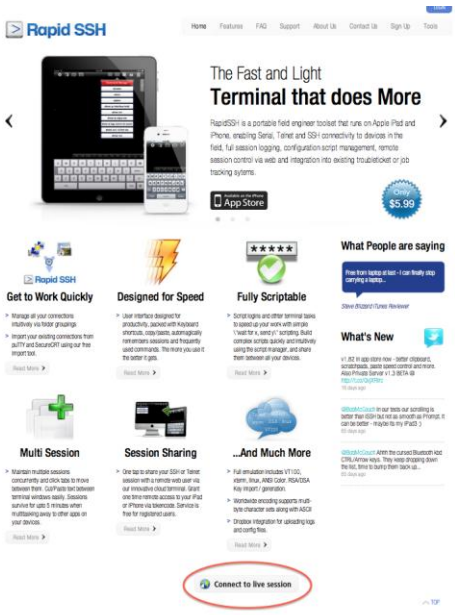


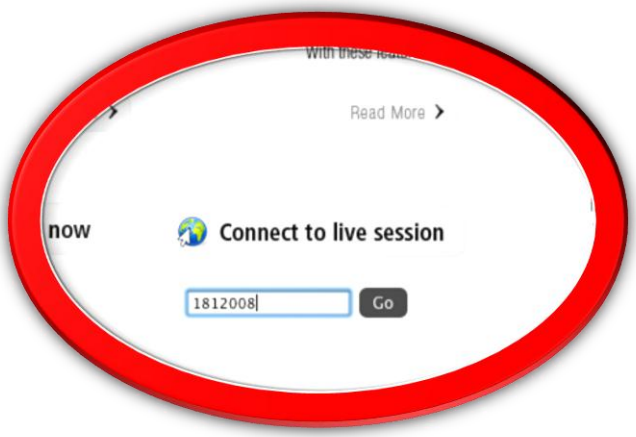
4.4.2 Remote User Access

Remote access to shared terminal sessions is dependent on whether they have been shared with the RapidSSH.com public servers or via the users own privately hosted server (Private Server). For the

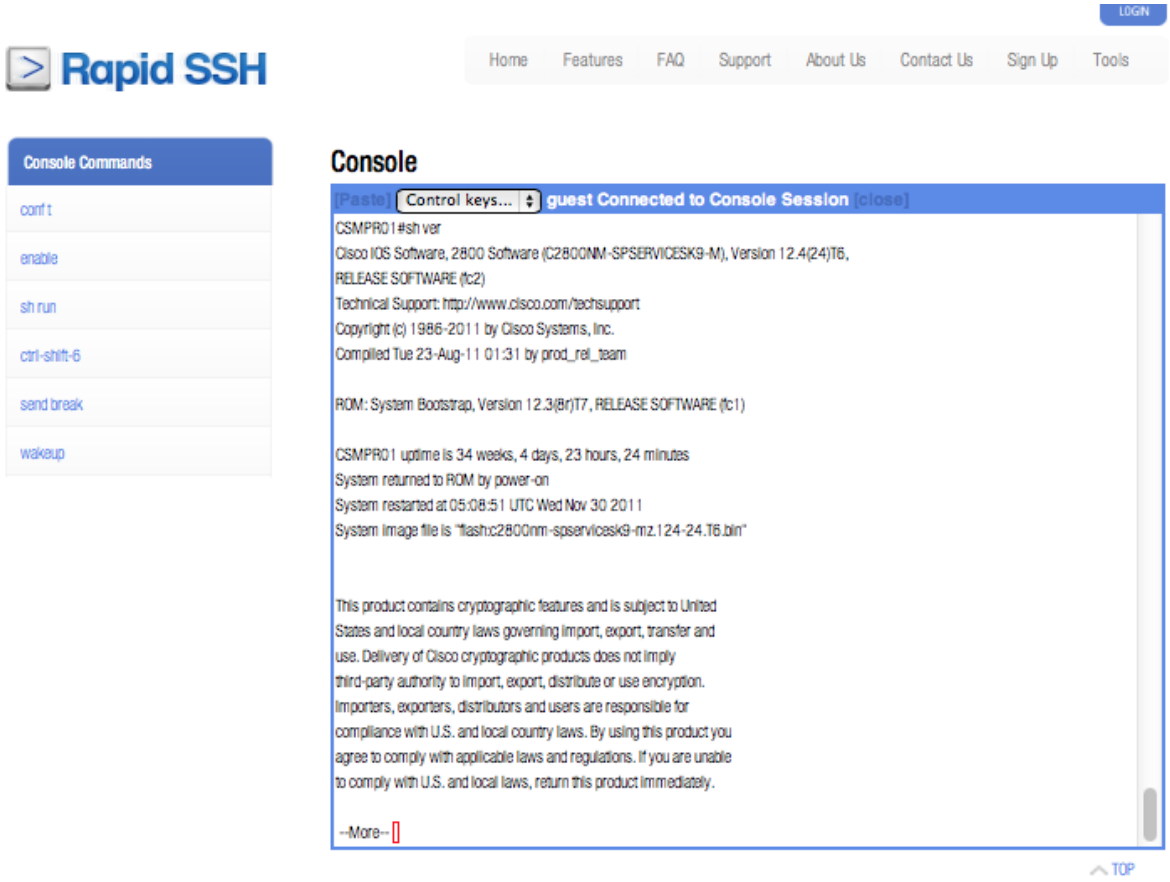
purpose of this manual it is assumed shared with the RapidSSH.com public servers. To learn more about RapidSSH Private Server visit www.RapidSSH.com/private-server

When the remote user has obtained the one-time token code, they enter it on the home page at www.RapidSSH.com, or if the token code was shared with the remote user via email there is a single link to click in the email to directly connect.





After entering the token code, the code is checked against currently active and available codes, and if correct the RapidSSH web terminal will launch. The remote user is presented with a plugin free web terminal that mirrors what is on the screen of the RapidSSH app running on the iPad/iPhone.


LOGIN
Rapid SSH
Home
Features
FAQ
Support
About Us
Contact Us
Sign Up
Tools
Console Commands
conf t
enable
sh run
ctrl-shift-6
send break
wakeup
Console
[Paste]
Control keys...
guest Connected to Console Session
[close]
CSMPR01#sh ver
Cisco IOS Software, 2800 Software (C2800NM-SPSERVICESK9-M), Version 12.4(24)T6,
RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2011 by Cisco Systems, Inc.
Compiled Tue 23-Aug-11 01:31 by prod_rel_team
ROM: System Bootstrap, Version 12.3(8r)T7, RELEASE SOFTWARE (fc1)
CSMPR01 uptime is 34 weeks, 4 days, 23 hours, 24 minutes
System returned to ROM by power-on
System restarted at 05:08:51 UTC Wed Nov 30 2011
System image file is "flash:c2800nm-spservicesk9-mz.124-24.T6.bin"
This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.
--More--
TOP

Both iPad/iPhone user and remote user can interact with the terminal session at the same time, however only one remote user can access any given session. If a later remote user enters the same token code, their later session will take over from any existing remote user.

Note that the remote user can only interact with the terminal while the RapidSSH app is foreground on the iPad/iPhone. If RapidSSH is in the background, then it will still maintain its Telnet/SSH session for upto 10 minutes, however the remote user will not be able to interact with the session until RapidSSH is brought to the iPad/iPhone foreground again.

4.4.3 Stop Sharing Session

To stop sharing an iPad/iPhone terminal session with a remote user, tap the button to activate the Session Manager popover, then tap the button for the terminal tab that sharing should be disconnected. The world icon will turn grey and the subtitle will change from the token code to say “Not Shared”.

5 Terminal Features

This section discusses the following RapidSSH terminal features:

- Keyboard popup bars
- Command Shortcuts
- Password Shortcuts
- Clipboard Viewer

5.1 Terminal Features

5.1.1 Keyboard Control

Use the button to show and hide the onscreen iPad keyboard. Used to increase screen real estate, especially useful for debugging in the iPhone/iPod with its smaller screen. Note that when a Bluetooth keyboard is connected the software keyboard is no longer visible.

5.1.2 Keyboard Popup Bar Selector

Pressing the

button changes or hides semi translucent popup bar of shortcuts that sits above the keyboard. Hiding the bar can be useful when trying to read the line at the bottom of the screen. Pressing this button cycles through some preconfigured popup bar options.

Options:
- **CISCO – keys useful when working on Cisco routers and switches**
- **SUN – keys useful when working on Sun Servers**
- **F1-F12 – Function Keys**
- **Custom – user selectable keys**
- **None – Popup bar disappears allowing full visibility of the bottom line of the terminal**

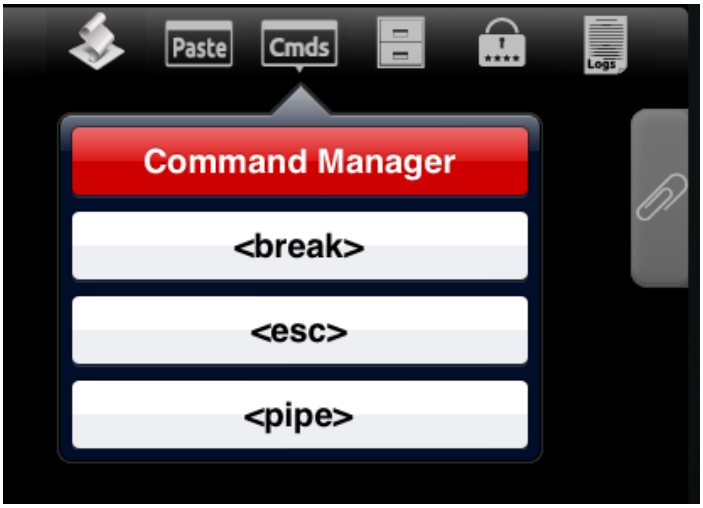
For the Default “Custom” option, the user can create their own combination of buttons on the popup bar by pressing and holding each individual button. A button selection popup will allow these keys to be changed to one of many options. In version 1.81 or later any key from the Apple keyboard can be installed on the Custom keyboard bar, in addition to the ones available in the popup picker. Using the CTRL key on the picker prior to selecting a Apple keyboard key will put CTRL-[selected key] instead of the normal key.

TAB ~. #. ESC ^^
F1 F2 F3 F4 F5 F6 F7 F8 F9
F10 F11 F12 CTRL BRK PASTE Clear Screen Clear Scroll
Pick a key to replace (from keyboard or above)
Ctrl
ESC TAB CTRL ^A ^E ^F ^G
Q W E R T Y U I O P
A S D F G H J K L return
Z X C V B N M
.?123
.?123

5.1.3 Command Manager

Command shortcuts are added in the Command

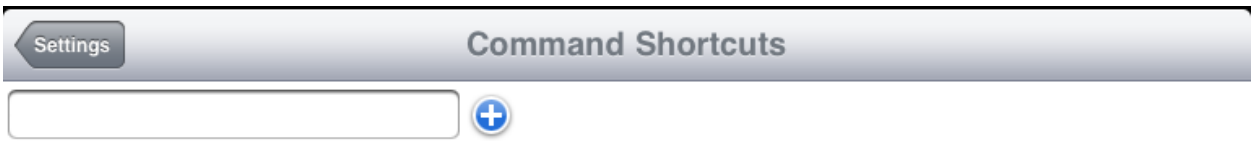
Manager. Once entered they are available with one tap via the “Cmds” popup/dropdown from the top navigation bar. Each command is sent with a carriage return after it.



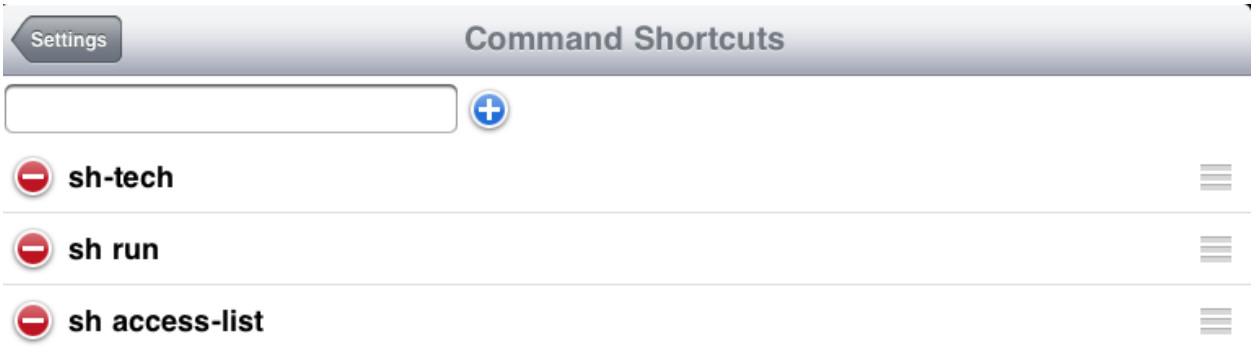
Tapping CMDS will show the list of current command shortcuts. A few shortcuts have already been defined by default and user can add their own custom command shortcuts for use in the terminal

Defaults:
- **Command Manager (Launches Command Manager page for entering more quick commands)**
- **<break> - sends terminal break sequence**
- **<esc> - sends ESC sequence for selected encoding method**
- **<pipe> - sends the | symbol WITHOUT a Carriage Return**

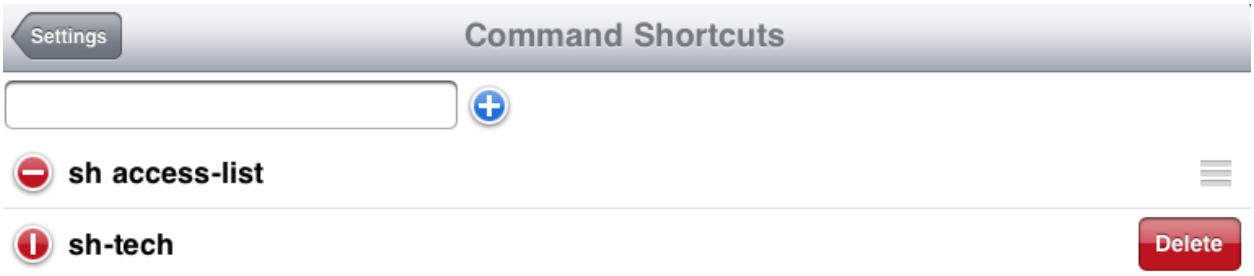
Adds command shortcuts here for use in terminal via dropdown icon



Rows on the right: Hold button to rearrange the order of the command shortcuts as they appear in the dropdown popover.



Stop sign on the left: Changes the rows on the right to delete button which then can delete the saved command shortcut



5.1.4 Password Manager

Like the Command Manager, Allows user to save passwords to be used as shortcut instead of typing the entire password all over again

Add passwords here for quick use in terminal via dropdown icon

Saved passwords are recalled in terminal sessions by tapping the button. While the passwords as entered are visible in the Password Manager, from the terminal window the middle characters are obfuscated when recalled.

5.1.5 Clipboard Viewer

The iPad version of RapidSSH has a slide out clipboard window accessed via the paperclip icon. This allows the user to see and edit the contents of the iPad's clipboard and then paste

The iPhone version of RapidSSH has a whole page clipboard viewer with the same buttons for pasting to the terminal window.

5.1.5.1 Paste Clipboard to Console

Pastes the content of the clipboard to the console. If the user has made changes to the clipboard viewer text, but has not selected and copied the clipboard viewer to the internal clipboard, then this button will paste the clipboard viewer contents *prior* to any edits.

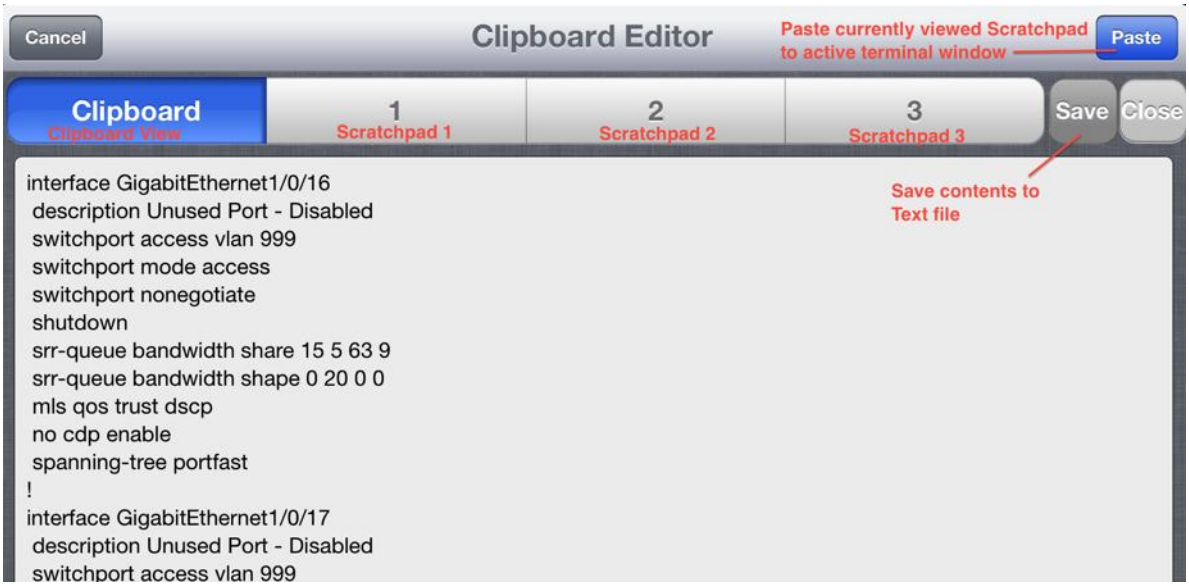
5.1.5.2 Paste Editor to Console

Pastes whatever is in the editor (the box containing the content of the clipboard) to the console. This allows the user to modify the content of the clipboard first before pasting to the console. Effectively what this button does is copy the edited view to the clipboard prior to then pasting to the terminal window.

Once the editor is closed, the content of the clipboard will revert to the original clipboard again unless the edited "Editor" is copied which then replaces the clipboard content.

5.1.5.3 Launch Scratchpads

RapidSSH v1.82 and above has 3 additional "scratchpads" that can be used by user for loading in text files for editing and pasting to the active terminal session.



Selecting an unused Scratchpad can be used to either create new text or load a file into for editing. Selecting cancel returns to the active terminal window.

6 RapidSSH File System

6.1 RapidSSH File Types

RapidSSH stores files within a portion of the iPad/iPhones file system that is dedicated to the RapidSSH App. There are 5 types of files that are stored:

- User created text based files that are typically used in a terminal session via cutting/pasting into the terminal window via the Clipboard Viewer.
- Log files, which are RapidSSH generated logs for each terminal session that has logging enabled.
- RSA Keys for use in certificate based authentication in SSH. These files are imported via the Main Settings -> Private Key dialog box.
- ".connection" files which are saved connections visible in the Connection Manager. These files are created either in the RapidSSH Connection Manager, imported by RapidSSH File Manager, or imported via iTunes.
- ".script" files which are saved scripts that can be executed on login or during a terminal session via the Script Manager. These files are created either in the RapidSSH Script Manager, imported by RapidSSH File Manager or imported via iTunes

6.2 User Created Text Files

While there are 5 types of files, the File Manager is concerned only with

a) the text files type – User Created text based files.
b) .connection files – saved connections that can be imported into Connection Manager
c) .script files – saved terminal scripts that can be imported into the Script Manager

Other types of files are created / viewed / deleted / uploaded via their respective managers as discussed below.

User Created text files are typically used in terminal sessions – they could be snippets of configurations or other text based templates that are edited in the clipboard viewer or scratch pads before being pasted into the Terminal window.

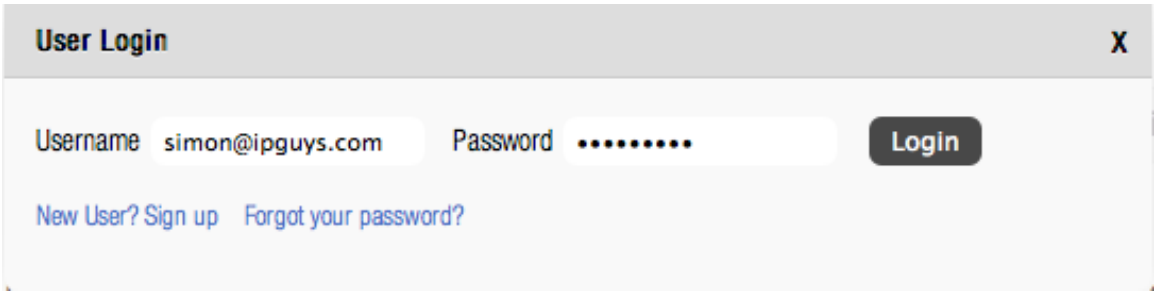
To be used by RapidSSH, the files must be local to the iPad/iPhone device. RapidSSH offers 2 ways to get User Created text files into the local file system:

- via import from the users portal on the www.RapidSSH.com website
- via import from the users /My Apps/RapidSSH/ folder in their Dropbox.com cloud storage

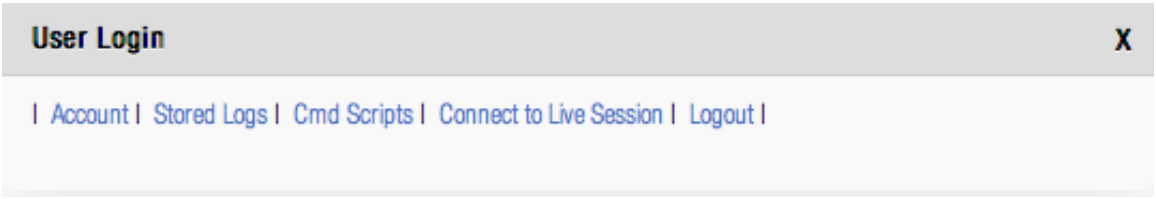
6.2.1 Importing files from RapidSSH.com

To import text files from the RapidSSH.com portal the user must first have signed up to a free account at www.RapidSSH.com/signup . After sign up, the user logs in via the Login panel at the top of any webpage.

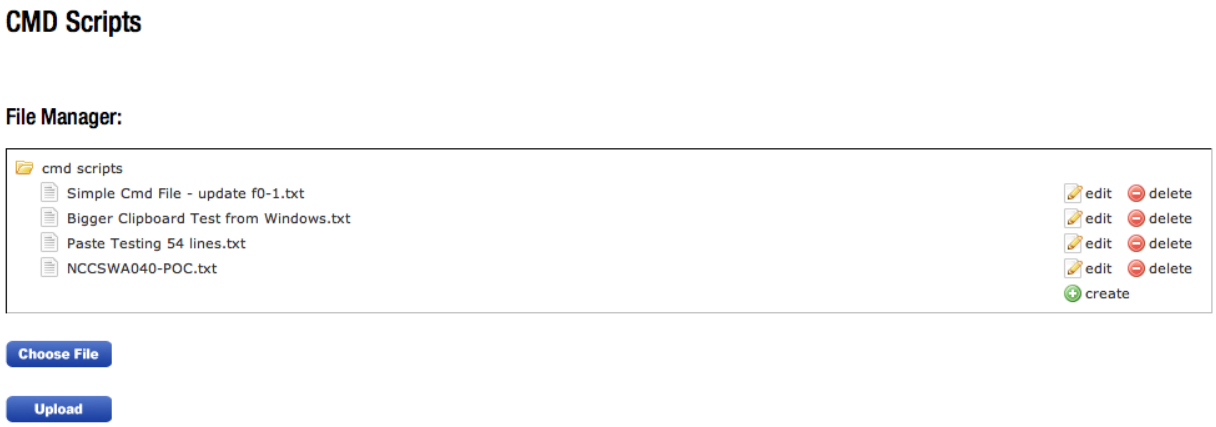
6.2.1.1 Creating / Uploading files on RapidSSH Website



Once signed in the website “Login” button changes to an “Account” button. Clicking on this reveals the users portal options:



User Created Text files (including .script and .connection files) are accessed via the “Cmd Scripts” button. This button will be renamed “User Created Files” in a later release of the website.

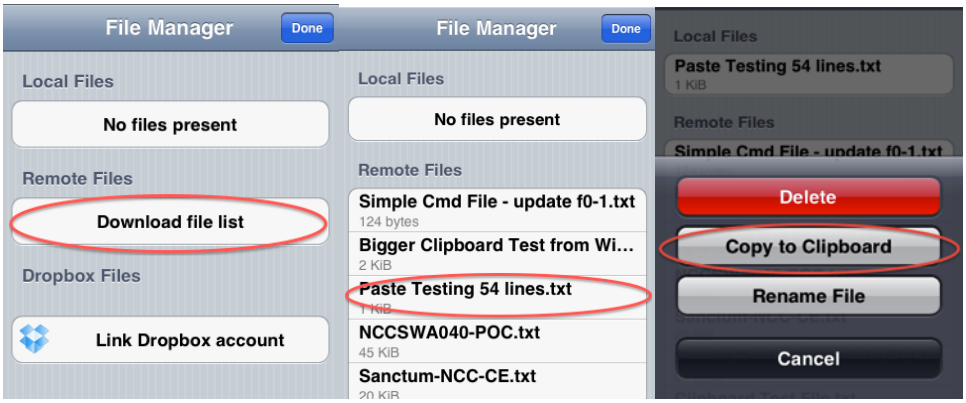


From the CMD Scripts user portal page, the user can upload, edit or create text files that will be visible to the iPad/iPhone RapidSSH application under the

6.2.1.2 Downloading Files from RapidSSH Website

Once files are uploaded to the users RapidSSH.com portal page, they are visible via the File Manager by tapping the "Download File List" button. The users RapidSSH.com credentials (email / password) must be entered into the Main Settings -> Session Sharing field as described in section 3.6 above in order for the remote files to be visible.

The list of files is presented to the user. Tapping on one of the remote files provides option to download to the local store. Once in the local store tapping the file name again allows for the file to be copied to the clipboard viewer for editing/pasting, renaming or deleting. If the file tapped in the local store is a .script or .connection file, the RapidSSH pop-up will allow these files to be imported into the Script or Connection manager respectively.



6.2.2 Importing files from Dropbox account

To import text files from a dedicated RapidSSH folder within the users Dropbox.com account, the user must first link their Dropbox account with the RapidSSH app. After successfully linking, the File Manager will be able to see User Created text files stored in the users Dropbox/Apps/RapidSSH folder, and download them to the iPad/iPhones local storage.

Once in local storage the same operations are permitted as described for RapidSSH portal files are possible (Copy to Clipboard, Rename or Delete).

6.2.2.1 Linking Dropbox Account to RapidSSH

From version 1.82, linking to Dropbox is performed from the Main application settings. In the main settings, tap the Link to Dropbox account button. This will start the mobile safari browser where the user will enter their dropbox.com credentials to login, or alternatively, if they already have the Dropbox iOS application installed on their iPad/iPhone it will present a simple allow/disallow dialog box. Tap Allow, after which control will return to RapidSSH. From RapidSSH File Manager, it will now be possible to download Dropbox folder contents to the local storage area.

Note that once linked, the user can unlink RapidSSH from their Dropbox account either within the Main settings or via the Dropbox website. Only a single folder is linked to the users Dropbox repository (not including any subfolders) therefore any files the user wants visible to the File Manager must be in this folder to be accessible.

6.3 Log Files

RapidSSH offers comprehensive logging of individual terminal sessions. Once logging is enabled in the main settings, all printable screen output is captured to a logfile.

One log file is created for each session. Unless a script is being used to append a log file, a new log file is created each time a session is stopped and restarted.

On iPad Logs are accessed via the button, whereas on iPhone Logs are accessed via and then "Logging"

6.3.1 Log File Naming

For connections that are started with the "Quick Connect" method the log file name will be:

Log_YYYY-MM-DD_HHMMSS.txt

For connections that are started from a saved connection in the Connection Manager, the file name will instead be appended with the connection name followed by date: ie

Server1_YYYY-MM-DD_HHMMSS.txt

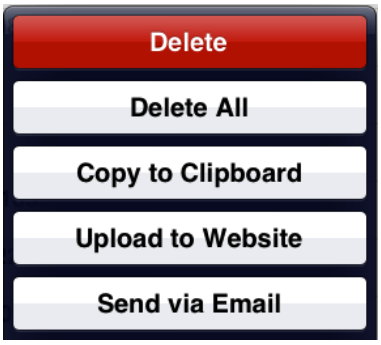
Invoking log operations from a script can change the name of the log files. See the Script Manager section above for details on log operations that are possible via scripting.

6.3.2 Uploading Log Files

Log files can be extracted from the iOS device via 2 possible methods:

- Via upload to the users portal page on the RapidSSH.com website
- Via upload to the users Dropbox folder dedicated to RapidSSH (default is Apps/RapidSSH)
- Via email of individual log files from the iPad/iPhone mail client

Navigate to the Log Files page, then tap a log file name. Select the one of the 3 upload options. Note to upload to Dropbox, the users Dropbox account must be linked to the RapidSSH app as described in section 6.2.2.1 above.



Upload to RapidSSH website user portal is invoked via the "Upload to Website" button. As per the File Manager, the user must have a (free) RapidSSH.com account and the credentials entered as described in section 3.6 above.

Access to uploaded logs on the RapidSSH.com website is via the Stored Logs button.

6.3.3 Auto-Uploading Log files

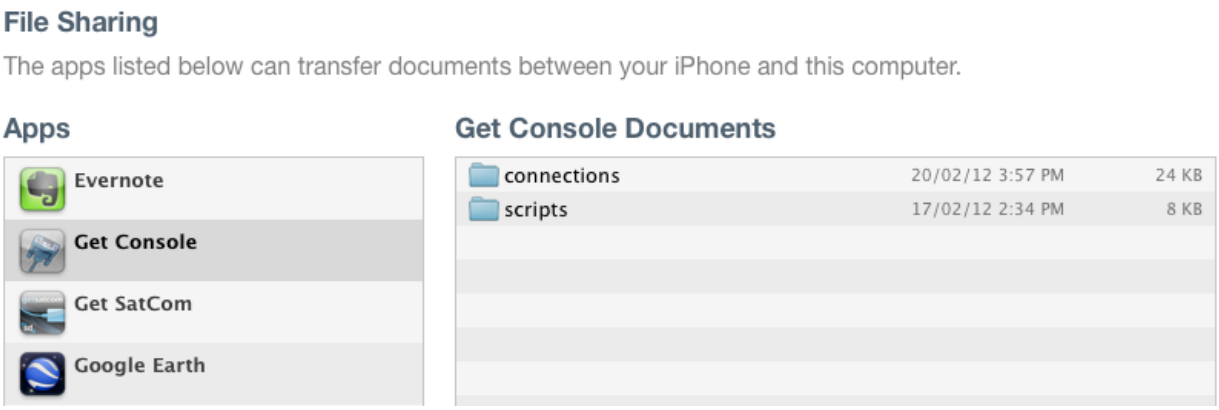
As discussed at section 3.1.8 RapidSSH can be configured to automatically upload log files on session completion to either Dropbox or to the users portal page on the RapidSSH website. To enable Auto-Upload, select the appropriate option from the Main Settings.

Note that when Auto-Upload logs is enabled, the iPad / iPhone must have Internet connectivity via 3G or Wifi at the completion of the terminal session. If there is no Internet connectivity, then the auto-upload will fail, and RapidSSH will NOT re-attempt to auto upload that log file. The log file will still be saved locally and available for manual upload.

6.4 .Script and .Connection RapidSSH Files

RapidSSH stores .connection and .script files for saved connections and saved terminal scripts respectively. These are created and managed in the Connection Manager (see section 4.2) or Script Manager (see section 3.1.9.6). These are simple XML files and can be manipulated externally from the RapidSSH application and then re uploaded.

Existing .connection and .script files can be extracted via iTunes or iPhone Explorer (see section 4.2.3). To extract via iTunes, connect iPhone/iPad to PC/Mac, run iTunes, navigate to the iPad/iPhone device and then select Apps.



The .connection and .script files can be extracted via the "save to…" button at the bottom of the page.

Individual .connection and .script files that have been created or edited externally can be uploaded via the "Add…" button. When RapidSSH starts it checks the root folder for .script and .connection files and moves these into the correct internal folder.

From version 1.82 and later, .connection and .script files are detected via the local file system and when tapped in the file system viewer are optionally installed into Script Manager or Connection Manager automatically. It may be necessary to stop and restart the RapidSSH application on the iPad/iPhone to trigger the auto-installation of connections.zip files.

7 General Troubleshooting

This section covers common problems reported by users, and typical fixes / workarounds

7.1 Session Sharing Issues

Session Sharing to RapidSSH Public Servers fails. If user receives "Failed to connect / read from remote control server":

- Check your internet connection (can you browse in mobile safari to www.RapidSSH.com)
- Check Session Sharing settings configurations defined in Main Settings -> Session Sharing are correct (see section 3.6).
- If everything is correctly configured, try again – depending on latency the first attempt to connect can fail due to underlying encryption key exchan2ge taking too long. The next attempt works as the encryption keys are cached by the RapidSSH app.
- If still having issues with a public RapidSSH server, try to use an alternative one (ie Asia Pacific rather than North America) .